
# MSM8960 Thermal Mitigation Algorithm

***80-N8633-1 C***

***November 21, 2012***

---

**Submit technical questions at:**
**https://support.cdmatech.com/**

**Confidential and Proprietary – Qualcomm Technologies, Inc.**



**Qualcomm Technologies, Inc.**
**5775 Morehouse Drive**
**San Diego, CA 92121**
**U.S.A.**

# Contents


1 Introduction .......... 6
1.1 Purpose .......... 6
1.2 Scope .......... 6
1.3 Conventions .......... 6
1.4 References .......... 7
1.5 Technical assistance .......... 7
1.6 Acronyms .......... 7

2 Purpose of Thermal Mitigation .......... 8
2.1 Overview of thermal issues .......... 8
2.2 Thermal management physical design and layout techniques .......... 9
2.3 Thermal management algorithm goals .......... 9
2.4 Thermal management algorithm prerequisites .......... 10

3 Thermal Characterization .......... 11
3.1 Calibration of a specific form factor (phone or dongle) .......... 11
3.1.1 Purpose of thermal calibration .......... 11
3.1.2 Variations requiring change in calibration .......... 11
3.1.3 Calibration methods .......... 12

4 Thermal Software Architecture for Android .......... 15
4.1 Boot thermal mitigation .......... 17
4.2 KTM .......... 18
4.3 ThermalD .......... 18
4.4 Overall AP Thermal Management .......... 19
4.5 Thermal sensors .......... 20
4.5.1 TSENS .......... 20
4.5.2 TSENS calibration .......... 20
4.5.3 HKADC .......... 20
4.6 TMD .......... 21
4.7 AP-side thermal mitigation configuration .......... 21
4.7.1 Thermal zone definition .......... 21
4.7.2 Configuration file requirements .......... 24
4.7.3 Configuration file example .......... 25
4.7.4 Editing configuration file .......... 26
4.7.5 ThermalD logging .......... 26
4.8 Modem thermal mitigation .......... 26
4.9 Modem thermal mitigation feature set .......... 27
4.10 Modem-side thermal mitigation configuration .......... 28

4.10.1 Common NV settings ..... 28
4.10.2 LTE technology-specific NVs/EFS ..... 28
4.10.3 WCDMA/HSPA technology-specific NVs/EFS ..... 31
4.10.4 1X/DO technology-specific NVs/EFS ..... 31

**5 Modem Thermal Mitigation Details ..... 32**

5.1 LTE thermal mitigation functioning ..... 32
5.1.1 Uplink flow control and Tx backoff behavior ..... 32
5.2 WCDMA/HSPA thermal mitigation ..... 37
5.2.1 RLC data throttling and Tx backoff behavior ..... 37
5.3 1X/DO thermal mitigation functioning ..... 37

**6 Thermal Management Test Results ..... 38**

# Figures

Figure 3-1 Measuring the temperature profile ..... 13
Figure 4-1 LA thermal software architecture ..... 15
Figure 4-2 TSENS sensor locations ..... 16
Figure 4-3 Boot and kernel thermal mitigation ..... 17
Figure 4-4 Overall AP TM mechanism ..... 19
Figure 5-1 Uplink flow control ..... 33
Figure 5-2 UL flow control ..... 34
Figure 5-3 UE states – Mitigation 1 ..... 35
Figure 5-4 UE states – Mitigation 2 ..... 36

# Tables

Table 1-1 Reference documents and standards ..... 7
Table 4-1 Thermal zone ..... 21
Table 4-2 Sensor names ..... 22
Table 4-3 Sensor actions and action information ..... 23
Table 4-4 Thermal mitigation feature set ..... 27

# Revision history

| Revision | Date | Description |
|---|---|---|
| A | Dec 2011 | Initial release |
| B | Jan 2012 | Updates for thermal daemon and configuration file |
| C | Nov 2012 | Numerous changes were made to this document; it should be read in its entirety. |

# 1 Introduction

NOTE: Numerous changes were made to this document. It should be read in its entirety.

## 1.1 Purpose

As Qualcomm Technologies, Inc. (QTI) chipsets continue to advance, with heightened capabilities, performance, and concurrency, there is a need to have a thermal management framework in place to deal with high-power dissipation use cases. The thermal issue manifests in cases where temperatures of devices approach or exceed Underwriters Laboratories (and other worldwide regulatory bodies), carrier, and customer requirements. The primary ways of dealing with the increased power requirements and associated heat is, first, efficient power management as described in [Q4] and [Q5], and second, mechanical designs that efficiently dissipate the heat created as described in [Q2] and [Q3]. In addition, QTI provides thermal mitigation algorithms to deal with those instances where the temperature exceeds the desired operating levels. The software-based thermal management algorithm reduces the thermal load on the system by throttling performance of various features.

This document describes the mechanisms necessary for customizing the thermal management algorithm on devices using supported QTI ICs. This document also includes a brief introduction to threshold selection using thermal characterization, and describes proper configuration of the algorithm.

## 1.2 Scope

This document is intended for all engineers who use MSM8960 chipsets.

## 1.3 Conventions

Function declarations, function names, type declarations, and code samples appear in a different font, e.g., `#include`.

Shading indicates content that has been added or changed in this revision of the document.

## 1.4 References

Reference documents are listed in Table 1-1. Reference documents that are no longer applicable are deleted from this table; therefore, reference numbers may not be sequential.

**Table 1-1 Reference documents and standards**

| Ref. | Document | |
|---|---|---|
| ***Qualcomm Technologies*** | | |
| Q1 | *Application Note: Software Glossary for Customers* | CL93-V3077-1 |
| Q2 | *Application Note: Thermal Design Considerations* | 80-VU794-5 |
| Q3 | *Application Note: Thermal Design Checklist* | 80-VU794-21 |
| Q4 | *Presentation: Resource Power Manager (RPM) Overview and Debug* | 80-VP169-1 |
| Q5 | *Presentation: MSM8x60/APQ8x60 Linux Power Management Debugging Guide* | 80-VR629-2 |
| ***Standards*** | | |
| S1 | *Information Technology Equipment – Safety* | IEC 60950-1:2005 |
| S2 | *Safety Specification* | UL1905 |

## 1.5 Technical assistance

For assistance or clarification on information in this document, submit a case to QTI at https://support.cdmatech.com/.

If you do not have access to the CDMATech Support Service website, register for access or send email to support.cdmatech@qti.qualcomm.com.

## 1.6 Acronyms

For definitions of terms and abbreviations, see [Q1].

# 2 Purpose of Thermal Mitigation

The thermal management algorithm provides a safety net, preventing the user device from causing damage to components or injury/discomfort to users. To accomplish this, one must lower power dissipation to keep the temperature at acceptable levels. The algorithm carefully monitors key temperatures within the device and can be configured for the implementation of appropriate responses at the modem and Apps Processor (AP) level for MSM™ products, achieved via multiple management steps.

Finding the appropriate thermal performance tradeoff requires thermal calibration of the device. This is achieved by executing different combinations of scenarios on the device.

With more capabilities (such as higher DL and UL data rates), more powerful APs, and thermally hostile environments, the use of a thermal management algorithm becomes increasingly important to avoid any damage to components/impact to users while achieving carrier thermal performance standards.

## 2.1 Overview of thermal issues

Present thermal analysis shows several potential causes driving the need to manage thermal:

- Market demand for feature-rich devices increases thermal susceptibility.
- Multimedia-centric features, e.g., SVLTE, SVDO, Mi-Fi, multicore GHz CPUs, graphics processors, and HD video increase thermal dissipation.
- Smaller/thinner designs lead to smaller PCBs, leading to concentrating power density.
- The Power Amplifier (PA) reaches maximum junction temperature in small form factor devices; this is dependent on component selection and maximum Tx power.
- Enabling higher UL data rates tends to cause higher Tx power, leading to more PA thermal dissipation.
- Touch temperature of the device’s external casing may cause customer concerns; industry norms can limit smartphone external temperatures to 45°C (typically, ambient + 20°C) and external USB dongle temperatures to 75°C per [Q4] and [S1].
- Silicon-level temperature must stay below the specified limit (typically, for logic 125°C, and memory 85 to 105°C).
- Devices embedded in a Printed Circuit Board (PCB) (inside the computer housing) are exposed to higher ambient temperatures and restricted convective cooling forces, creating an even more severe thermal environment.

## 2.2 Thermal management physical design and layout techniques

To limit the impact of the thermal environment on the device, several techniques are suggested and recommended:

- Perform thermal analysis techniques during product design and development stages.
- Investigate device power and thermal budget from the analysis to locate projected hotspots.
- Implement efficient thermal management methods in the device mechanical structure.
- Ensure PAs reside in a location where heat is easily dissipated.
- Extensive use of thermal vias is necessary along with thermal paths within the PCB, allowing for lower thermal resistance of the device.
- Ensure no other large power dissipating components reside near or on the other side of the board from the PA and the MSM.
- Use of thermal dissipating techniques in the industrial design is highly recommended.

**NOTE:** See [Q2] and [Q3] for more information on thermal design.

## 2.3 Thermal management algorithm goals

Thermal management is targeted to:

- Protect components from exceeding thermal design limits; if the limits are exceeded, the Quality of Service (QoS) can be degraded and components can be damaged
- Ensure compliance with external case and touch temperature requirements from customers, carriers, and standard organizations (Underwriters Laboratories, PCI Express, and user expectations)
- Minimize the risk of power-limit constraints
- Manage the thermal risk and tradeoffs during concurrent operations
- Allow limited customer tailoring of temperature thresholds and methods for power reduction

## 2.4 Thermal management algorithm prerequisites

To properly manage the thermal scenarios of a device, the temperature must be measured from different thermally active points. It is done by sensors already present on the chipset and by placing thermistors at thermally active points, e.g., PA, PMIC, battery, etc. Currently, the thermal algorithm only works on PA thermistors reading.

The following lists preferred sensor locations. Each of these sensors can usually be read by the housekeeping ADC (HKADC) contained in QTI chipsets.

- Thermistor in close proximity to the PAs – The PA is the primary component contributing to thermal increase in a device, easily dissipating well over 1 Watt in poor RF conditions.
- Thermistor or internal sensor in MSMs/MDMs – The use of larger processors (AP or modem processor) tends to increase the heat generation of this component.
- Separate third-party component – If an external heat-generating component, e.g., some designs where third-party AP is used, a thermistor should be thermally coupled or in close proximity.
- TCXO thermistor – In extreme thermal environments, the TCXO tends to be most susceptible to heating, which can lead to dropped calls and poor performance. Sensing this condition helps determine when to invoke the thermal algorithm.
- Other locations, e.g., battery, PMIC, etc.

QTI recommends placing thermistors near thermally active devices.

# 3 Thermal Characterization

To properly evaluate thermal management against user-perceived QoS, the device needs to be calibrated thermally to utilize the management algorithm. Understanding the relationship between the internal temperature sensors and the actual temperatures of the components and external surfaces greatly assists in proper configuration of the device.

## 3.1 Calibration of a specific form factor (phone or dongle)

Every form factor and hardware design iteration must have thermal testing performed for understanding of heating profiles, e.g., layout, size, and PCB characteristics, all of which impact thermal loading.

### 3.1.1 Purpose of thermal calibration

Characterizing the thermal behavior of the device:

- Calibrates the external thermistors with the components monitored; this helps establish the correlation of the thermistor temperature with the actual component of concern. This is done by characterizing the ADC reading and making changes in software as needed.
- Calibrates how measurable internal temperatures of a device map to external temperatures exposed to the customer; external temperatures are not measurable in the real customer device, therefore, understanding the equilibrium and transient thermal loads and how they impact the internal sensors and external temperature is important if external temperatures are a concern.
- Provides use cases and their thermal impact on components and customers; various concurrent scenarios may thermally load the device differently, and understanding the hotspots for each may be important in configuring the algorithm.
- Provides information on the transient nature of the device (thermal time constant); measuring the thermal time constant provides the required time to engage the thermal management algorithm. Once engaged, it determines how long the algorithm takes to stabilize and/or cool the device.

### 3.1.2 Variations requiring change in calibration

Thermal testing and modeling suggests a new board or new form factor requires recalibrating the thermal characteristics and tuning the algorithm. Changes in the placement of components, changes in the location of temperature sensors, and especially changes in form factor dimensions all change the thermal characteristics.

## 3.1.3 Calibration methods

### 3.1.3.1 Full commercial software performance execution

The use of final software to perform peak performance on both the modem and AP is the preferred method for thermally calibrating the device. Running benchmarks or using Factory Test mode can provide some utility, but the concurrencies experienced and the power dissipation levels experienced in the components are very different between real-world use cases and test modes.

### 3.1.3.2 Identifying high power dissipation thermal use case scenarios

Use cases producing the maximum thermal effects, which are needed to activate the thermal management algorithm, involve multiple sensor points in various components. This implies several of the following scenarios.

#### Modem side

- RF conditions – Poor RF link in a stationary situation, which causes high Tx power
- A scenario with significant UL data throughput drives Tx power higher, with larger data packets
- High DL data rates in conjunction with applications such as video

#### AP side

- GHz CPUs with multiple cores
- Graphics processors
- HD video
- Concurrency
- Camera operations
- WLAN use

These scenarios and concurrent operation of the scenarios tend to produce thermal effects requiring thermal management. Measurement of the temperature profile over several thermal time constants for each scenario, as shown in Figure 3-1, determines if:

- Steady state temperature will exceed acceptable limits of any component, or the external surface
- Components influencing the thermal response
- Steady state temperature requires a thermal response of:
  - Normal – Normal operating region where power reduction is not needed
  - Mitigation 1 – Operating region where first set of mitigation steps are performed; this can be reducing the power by using flow control if desired on the modem side, and reducing the CPU clock on the apps side
  - Mitigation 2 – Operating region where second step of mitigation steps are performed; this can be reducing the power further by using Tx backoff as desired on the modem side or reducing the CPU clock further on the apps side.
  - Emergency – Operating region where damage to components/users is imminent and hence drastic steps such as tearing down the call are taken

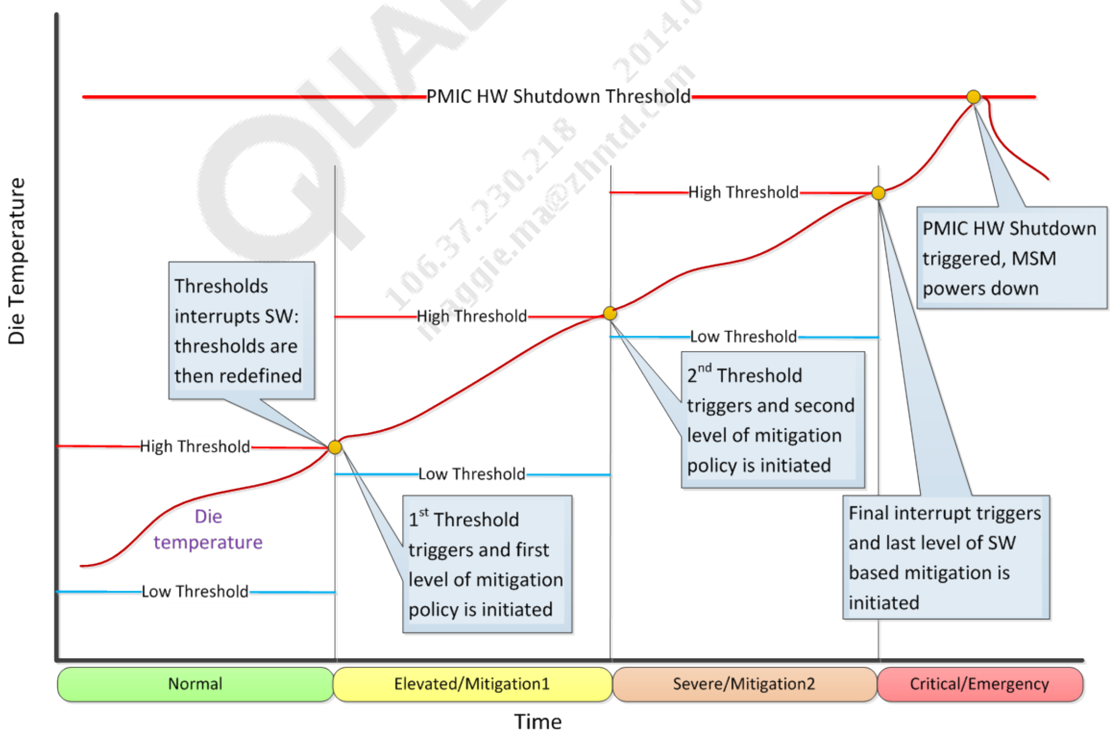


**Figure 3-1 Measuring the temperature profile**

#### 3.1.3.3 External heater use

If final commercial software is not available, thermal characterization of the form factor is possible by placing small patch heaters inside the device. Thermal loads of each patch heater should be representative of each component's power dissipation for each thermal scenario identified.

**NOTE:** Lab scenarios present several factors tending to over-represent the thermal response:

- Long durations of maximum Tx power are unlikely due to a variety of fading and coverage conditions on real networks.
- 100% UL data duty-cycle is unlikely due to nuances of the network, scheduling, and data packet readiness.
- Long durations of full DL bandwidth to a single user are unlikely due to the needs of the network to share bandwidth among many users.

The thermal realities experienced in the field tend to be much less severe than laboratory thermal conditions. While not limiting the potential thermal liability, real network conditions temper the plausible customer impact as witnessed during internal testing. Internal device temperatures in small form factors reach 85°C in the laboratory environment and seldom exceed 60°C in the field across many scenarios.

# 4 Thermal Software Architecture for Android

Thermal management software, as shown in Figure 4-1, consists of three key components, the thermal daemon, sensor inputs, and thermal management devices. In addition, there are algorithms for boot and kernel initialization to manage thermal performance.

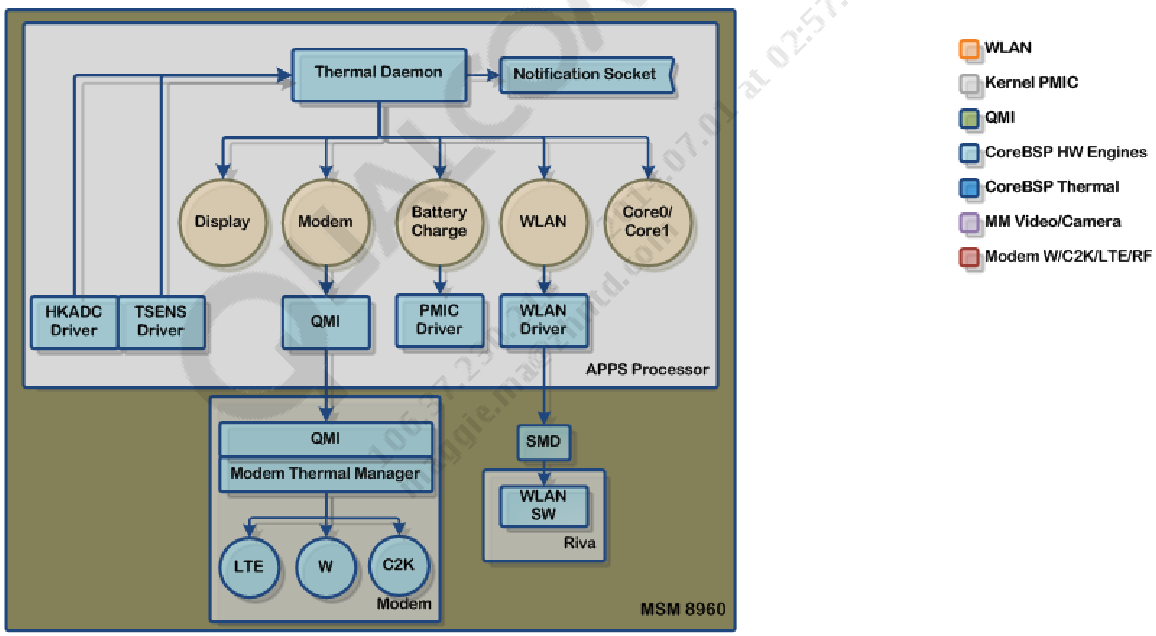


**Figure 4-1 LA thermal software architecture**

The thermal daemon, a super user process running in Linux Android™ user space, is the central control for thermal software. Sensor inputs for temperature readings come from two sources. Temperature sensors (TSENS) are embedded in five locations in the MSM, as shown in Figure 4-2. The second source is the HKADC in the PM8921 connected to the MSM8960, which has one PA thermistor connected on analog inputs. Thermal management devices are software components listening to the thermal daemon. The thermal daemon sends events to management devices to restrict performance to a predetermined level during times of high thermal load on the system.

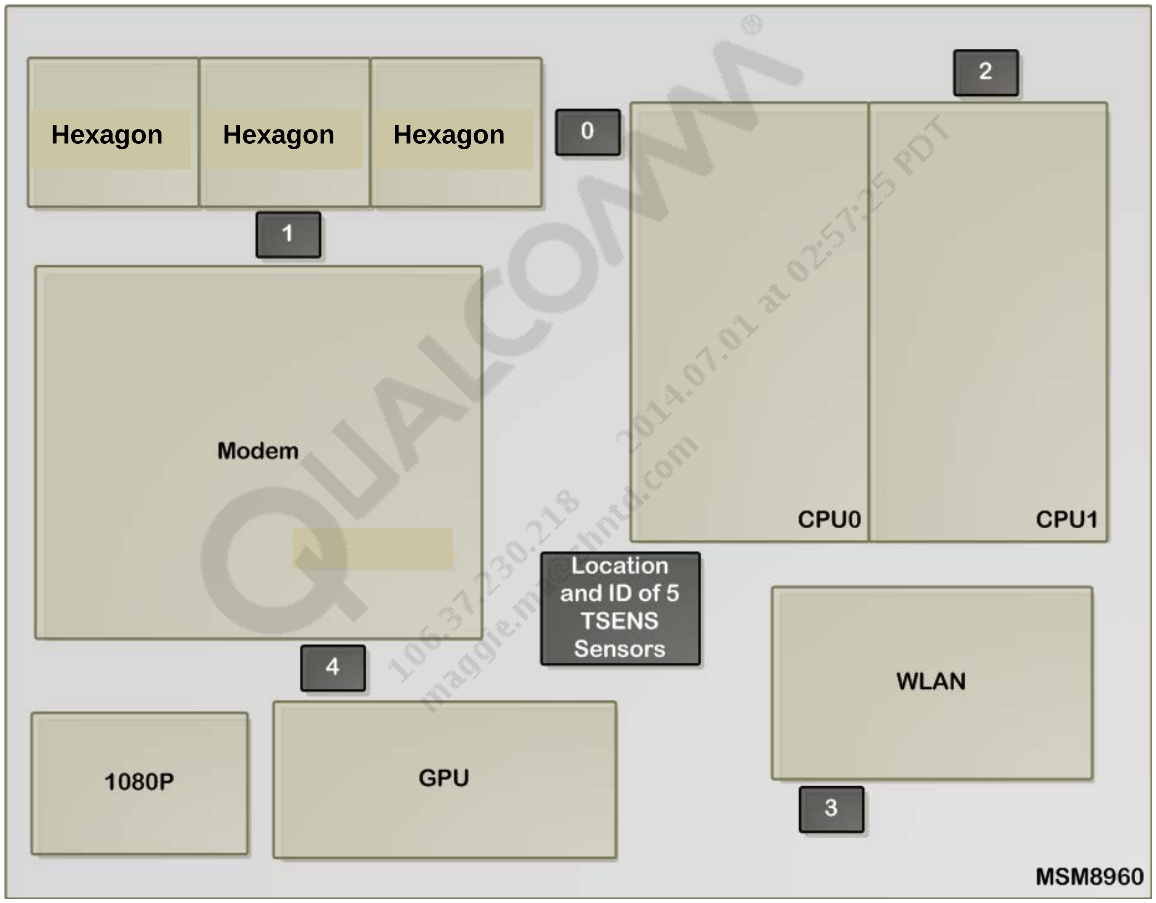


**Figure 4-2 TSENS sensor locations**

## 4.1 Boot thermal mitigation

Boot thermal mitigation ensures the device does not exceed temperature limits by adding a short delay if temperature is high prior to powering on high-power CPUs and initializing DDR. The Sensor 9 temperature is checked against a threshold set in the boot loader. If the temperature is above the threshold, a short delay occurs. After the delay the temperature is checked again. This repeats until either the maximum number of checks occurs or the temperature reduces below the threshold, as can be seen in Figure 4-3. The threshold, max number of check, and delay time are all configured in boot header files. Defaults for the threshold are set at 150ºC, effectively disabling this algorithm. However, some thermal designs may require this aggressive management technique to operate in some extreme environmental conditions. To change the defaults, just add these defines to the boot builds file with the desired values:

```
#define MAX_TEMP_CHK_ITER 40
#define MAX_WAIT_TIME_MICROSEC 250000
#define BOOT_TEMP_CHECK_UPPER_THRESHOLD_DEGC 150
```

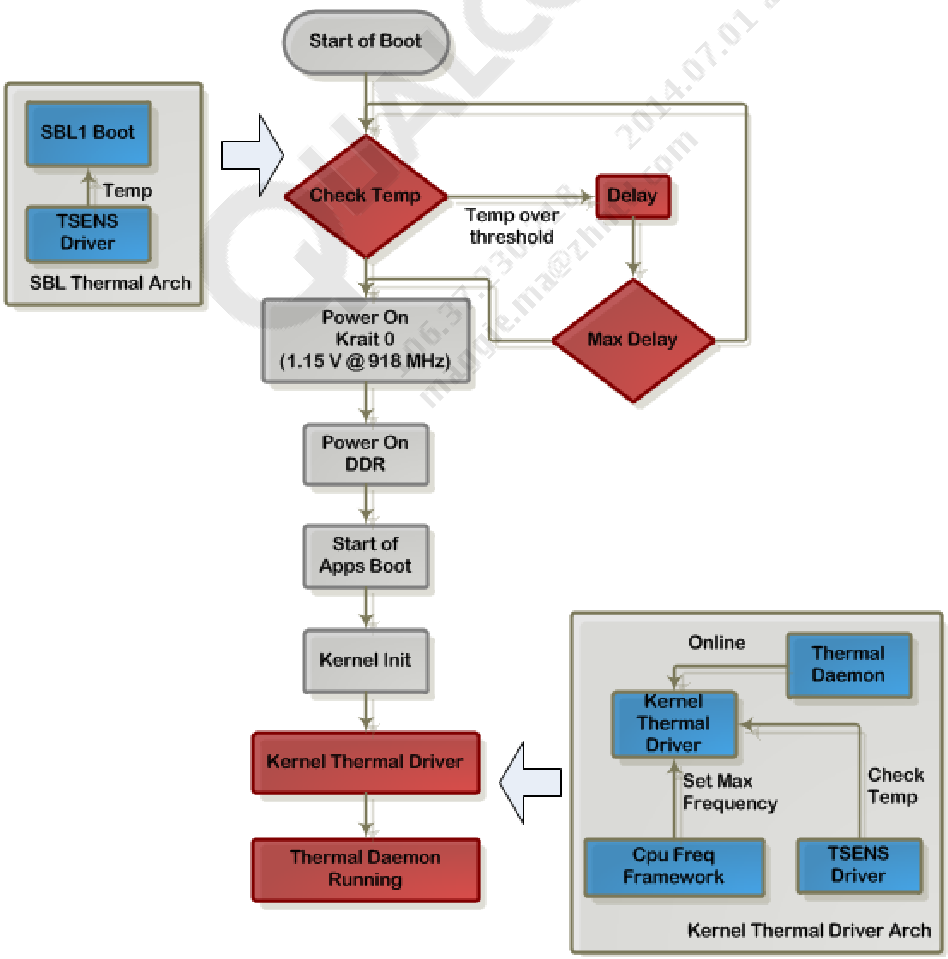


**Figure 4-3 Boot and kernel thermal mitigation**

## 4.2 KTM

The Kernel Thermal Monitor (KTM) provides thermal management during kernel initialization, prior to initialization of the main thermal management algorithm which requires services such as the file system to be operational. The KTM compares Sensor 0 to a preconfigured threshold every second. If the temperature exceeds the threshold, the performance level of the CPU cluster is reduced to a preconfigured operating point until the temperature reduces below the threshold. Once the thermal daemon process starts, a signal is sent to the KTM to stop checking the temperature.

## 4.3 ThermalD

The Thermal Daemon (ThermalD) user space process performs two main functions, monitoring temperature profile and controlling management devices.

Monitoring of the temperature profile is done by actions triggered at configurable temperature thresholds. On initialization, the ThermalD reads a configuration file containing all of the sensor and threshold data. If a temperature threshold is crossed, the ThermalD acts by sending events to management devices.

The interface to management devices, in general terms, is a management level. Typically, four levels are configured for any one of the devices. Level 0 is normal operation, Level 1 is the first level of management, etc.

In addition to the thermal management setup through the configuration file, the core control algorithm is hardcoded to ensure thermal protection. The core control algorithm monitors the temperature of each CPU core. If the temperature exceeds 115ºC, the core is taken offline through the hotplug mechanism in Linux. Once the core temperature reduces below 95ºC, the core control algorithm allows the core to come back online if required.

## 4.4 Overall AP Thermal Management

Three distinct AP thermal management schemes described in Section 4.1 through 4.3 are currently provided. After passing Boot Thermal Mitigation (BTM), boot loaders run with a single core at a speed of 918 MHz. This operating point is thermally safe in a typical mechanical design. When the Linux kernel initializes, both cores are started at maximum frequency. However, KTM does not start until a few services are initialized in the kernel. Consequently, there are a few seconds in which no thermal protection is operating while running the system at maximum CPU load.

In general this does not result in a thermal problem, but at extreme ambient temperatures in challenging mechanical designs this may not be sustainable over multiple back-to-back reboot cycles. In these rare cases, reduced initial CPU frequency or number of initial CPU cores used may be required to ensure thermal stability.

The KTM will operate until the user space thermal daemon is initialized. Overall timeline is depicted in Figure 4-4.

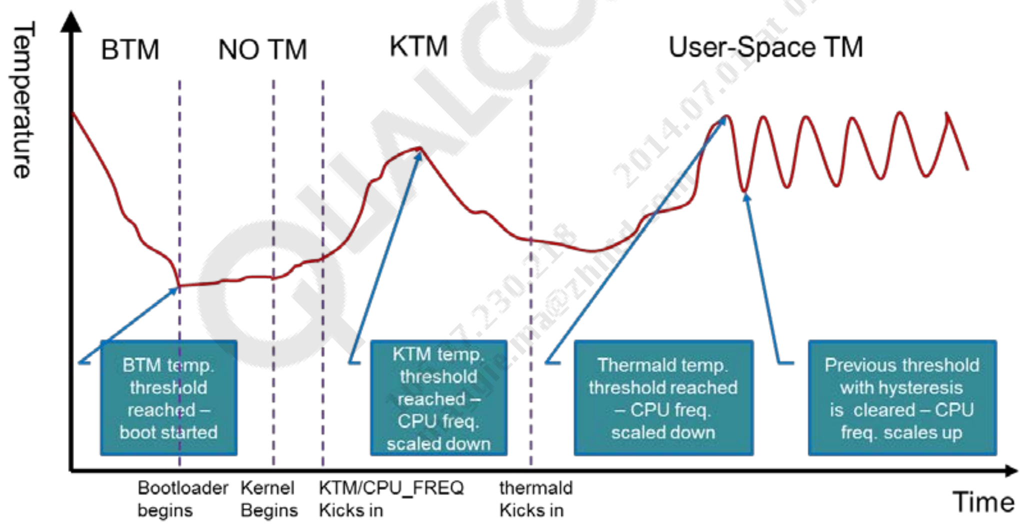


**Figure 4-4 Overall AP TM mechanism**

## 4.5 Thermal sensors

There are two sources for temperature data on the MSM8960 configuration, TSENS and HKADC reading from external thermistors. TSENS is a hardware block inside the MSM8960 with 10 temperature sensors connected. HKADC is a hardware block inside the PM8921 providing access to read analog inputs.

### 4.5.1 TSENS

TSENS hardware in the MSM8960 is controlled by a software driver running in Linux Android kernel space. No other processors access the TSENS hardware. There are 5 sensors, one main and 4 remote, which are monitored (see Figure 4-1). Each of the sensors inside the MSM8960 die is calibrated during chip fabrication. Upon initialization, the driver configures the TSENS hardware based on inputs from ThermalD and other self-contained data (such as calibration data). If calibration data is not available, the TSENS driver returns an error. The hardware is configured to continuously poll any sensors enabled at a programmable rate.

### 4.5.2 TSENS calibration

The calibration procedure writes an offset value into the QFPROM (fuse blow) for each sensor corresponding to the raw ADC reading at the ambient temperature at the time of calibration (typically 30°C). Additionally, the driver contains slopes for each of the sensors based on test data across many parts. The parameters are used by the driver to enable reporting of temperatures from each sensor in °C within the following accuracy limit of TSENS reading:

- 99.5% devices at 60°C – [-3.7°C, +4°C]
- 99.5% devices at 90°C – [-4.47°C, +4.9°C]

### 4.5.3 HKADC

The HKADC is a block in the PM8921 used to read the PMIC analog inputs. Some of the analog inputs are connected to thermistors providing temperature readings at hotspot areas outside of the MSM8960. On the MSM8960 reference design:

- One 99.5% device at 60°C – [-3.7°C, +4°C]
- 99.5% devices at 90°C – [-4.47°C, +4.9°C]

PA-based thermistors are connected to the PM8921. The readings from these thermistors are sent to the thermal algorithm through the QMI interface.

# 4.6 TMD

Thermal Management Devices (TMD) are software entities used to reduce the overall power consumption of a hardware block within the system, thereby reducing the thermal load. TMDs supported in the MSM8960 design are listed as part of the actions in Table 4-3.

# 4.7 AP-side thermal mitigation configuration

The thermal configuration file sets the operational parameters of the thermal management software. Each section, referred to as a thermal zone definition, is associated with one temperature sensor. Prior to the first thermal zone definition, the configuration file contains one line for the default sampling rate. The default is applied to any of the sensors without explicit sampling rates set in their thermal zone definition. Additionally, before the default sampling rate, inserting the word debug will enable verbose debug output from ThermalD.

## 4.7.1 Thermal zone definition

The thermal zone definition lists the configuration for one thermal zone consisting of six lines as shown in Table 4-1.

**Table 4-1 Thermal zone**

| Field name | Values | Units | Notes |
|---|---|---|---|
| temp sensor name | See Table 4-2 | | |
| sampling | 1000 to 65535 | ms | |
| thresholds | -273 to 273 | °C | Threshold to activate |
| thresholds_clr | -273 to 273 | °C | Threshold to clear |
| actions | See Table 4-3 | | Action to act upon at thresholds; multiple entries allowed are separated by + with no spaces; this should be same for all threshold levels |
| action_info | See Table 4-3 | | Information used by action; multiple entries allowed are separated by + with no spaces |

Temp sensor name listed in Table 4-2 corresponds to sensors located on the die shown in Figure 4-2.

Sampling is the rate at which the ThermalD will check the temperature against configured thresholds.

The final four fields, thresholds, thresholds_clr, actions, and actions_info, create a threshold group. There may be multiple threshold groups listed in each zone. Each threshold group defines a threshold to activate, a threshold_clr to deactivate, actions, and action_info associated with each action. Also, in each threshold group, multiple actions can be taken at the defined threshold. The entries must be separated by '+' with no spaces between the entries. Each action listed must also have a corresponding action_info separated by '+' with no spaces between the entries. Actions correspond to the available mitigation devices supported by thermal management as listed in Table 4-3. action_info is the level to be sent to the mitigation device listed in the associated action. The file is read during the ThermalD initialization to configure the thermal management system.

### Table 4-2 Sensor names

| Name | Notes |
|---|---|
| tsens_tz_sensor0 | Must always have at least one mitigation step enabled |
| tsens_tz_sensor1 | |
| tsens_tz_sensor2 | |
| tsens_tz_sensor3 | |
| tsens_tz_sensor4 | |
| pa_therm0 | GSM/UMTS/LTE PA |
| pa_therm1 | LTE PA – Applicable to only specific RF platforms |

## Table 4-3 Sensor actions and action information

| Action | Action info | Notes |
|---|---|---|
| none | Set to 0; not used but needed for parser to work properly | This mitigation step is disabled. If a threshold is crossed, no action is taken. |
| report | Set to 0; not used but needed for parser to work properly | The threshold crossing information is sent across an abstract local socket THERMALD_UI in a new line-separated string format. Parameters are sent in the following order:<br>▪ sensorname – Name of sensor reporting<br>▪ temperature – Current temperature<br>▪ current_threshold_level – Current threshold level triggered or cleared<br>▪ is_trigger – TRUE on level trigger, FALSE on level clearing |
| cpu | Sets the max allowable frequency for Krait CPUs | 0 to 2000000 kHz |
| gpu | Sets the max allowable frequency for GPU | This information will be provided in a future revision of this document. |
| wlan | Sends a mitigation request to WLAN to reduce allowable maximum Tx power | Levels 0 to 4<br>▪ 0 – No mitigation<br>▪ 1 – Disable aggregation<br>▪ 2 – Tx power backoff level 1<br>▪ 3 – Tx power backoff level 2<br>▪ 4 – IMPS |
| lcd | Sets the backlight intensity; will not adjust the max value over the currently configured brightness set by the user | 0 to 255<br>▪ 0 – Backlight off<br>▪ 255 – Maximum brightness |
| modem | Sends mitigation request to the modem on standalone design | Levels 0 to 3<br>▪ 0 – No mitigation<br>▪ 1 – Data throttling<br>▪ 2 – Tx power backoff<br>▪ 3 – Call drop/Maximum mitigation/Only E911 calls allowed |
| battery | Sends mitigation request to battery charging | Levels 0 to 3<br>▪ 0 – No mitigation<br>▪ 3 – Maximum mitigation |
| shutdown | Sends a shutdown request to the kernel with a delay to power down the phone | Delay in milliseconds |

### 4.7.2 Configuration file requirements

The configuration file must contain a thermal zone definition for tsens_tz_sensor0 as the first entry with the lowest of all temperature thresholds. The first threshold group in the tsens_tz_sensor0 thermal zone definition is used to set the TSENS threshold. Any other threshold groups in the tsens_tz_sensor0 thermal zone definition or threshold groups in any other thermal zone definition are only active after tsens_tz_sensor0 kicks in. For correct operation, the first threshold group in the tsens_tz_sensor0 thermal zone definition must have an activation threshold set lower than any other polling threshold in any other threshold group. This allows ThermalD to remain asleep until the first threshold is crossed, at which point ThermalD begins polling at a rate defined by the configuration file. Polling continues until all thresholds have cleared.

Also note that there should be a return character in the end of the thermald.conf file, i.e., entered at the last line of the thermald.conf file.

#### 4.7.2.1 Action field requirements

- The actions field lists the mitigation devices acted upon at each threshold. The parser for the configuration file is not complex, therefore, these fields are not completely free form. The rules for actions and actions_info within a threshold group must be separated by only the '+' character and no spaces.
- Actions must be explicitly stated for each threshold. Device settings are not implicitly reset to that of previous thresholds except upon clearing the lowest threshold.
- Each action must have a corresponding action_info.
- Action_info must also be separated by only the '+' character and no spaces.

### 4.7.3 Configuration file example

The example configuration file below turns on verbose debug output (line 1) and default polling rate to 5 sec (line 2).

Next, the first thermal zone definition is named tsens_tz_sensor0. The thermal zone definition sets a sampling rate for this sensor at 1 sec, which overrides the default 5 sec rate. The first threshold group in the tsens_tz_sensor0 thermal zone definition configures an activation threshold at 65°C, an action for the CPU to mitigate to a maximum frequency of 756 MHz, a second action for the LCD to mitigate to brightness level 200 out of 255; the threshold will clear at 60°C. The second threshold group configures an activation threshold at 75°C, an action for the CPU to mitigate to a max frequency of 540 MHz, a second action for the LCD to mitigate to brightness level 100 out of 255; the threshold will clear at 70°C. CPU and LCD management devices are reset to their default setting upon clearing the lowest threshold at 60ºC.

The second thermal zone definition is named pa_therm0. The thermal zone definition sets a sampling rate for this sensor at 1 sec, which overrides the default 5 sec rate. The first threshold group in the pa_therm0 thermal zone definition configures an activation threshold at 70°C, an action for the modem to mitigate to level 1; the threshold will clear at 65°C. The second threshold group configures an activation threshold at 80°C, an action for the modem to mitigate to level 2; the threshold will clear at 75°C. Modem management is reset to normal setting upon clearing the lowest threshold at 65ºC.

```
debug
sampling         5000
[tsens_tz_sensor0]
   sampling         1000
   thresholds       65          75
   thresholds_clr   60          70
   actions          cpu+lcd     cpu+lcd
   action_info      756000+200  540000+100

[pa_therm0]
   sampling         1000
   thresholds       70          80
   thresholds_clr   65          75
   actions          modem    modem
   action_info      1           2
```

### 4.7.4 Editing configuration file

To edit the configuration file, execute the following commands while connected to the device through a USB.

```
adb pull /etc/thermald.conf
adb remount
<edit>
adb push thermald.conf /etc/
```

### 4.7.5 ThermalD logging

ThermalD has minimal logging unless it is started in Debug mode or a debug flag is in the first line of the configuration file. Since ThermalD runs at startup, to turn on full logging, the daemon needs to be stopped and restarted.

```
adb shell stop thermald
adb shell
# thermald --debug &
```

The following command will output the ThermalD log information in the command window.

```
adb logcat -s ThermalDaemon
```

Timestamps can also be output by adding the –v time option.

```
adb logcat -v time -s ThermalDaemon
```

## 4.8 Modem thermal mitigation

Modem thermal management consists of three distinct mitigation levels:

- Uplink data throughput while keeping the original power class reduces the requirement of higher Tx power to close the reverse link.
- Downlink data throughput eases the power/MIPS requirements of the modem processor.
- Maximum Tx power limiting reduces the PA Tx power similar to SAR reduction.

Limited service mode restricts all data calls.

## 4.9 Modem thermal mitigation feature set

Table 4-4 lists the thermal mitigation feature set.

**Table 4-4 Thermal mitigation feature set**

| Feature | Radio technology | | | | Notes |
|---|---|---|---|---|---|
| | LTE | WCDMA/ HSPA | DO | 1X | |
| Thermal Emergency state; call teardown supervision and E911 allowance | Supported | Supported | Supported | Supported | |
| UL data throttling | Supported | Supported | Supported | Supported | |
| DL data throttling | NA | Supported | Supported | Supported | |
| Interference cancellation disabling | NA | NA | NA | NA | |
| Rx diversity disabling | NA | NA | NA | NA | |
| Tx power scaling – Duty-cycle backoff | Supported | Supported | Supported | Supported | Primarily used when data throttling is not effective or possible at edge of cell |

# 4.10 Modem-side thermal mitigation configuration

For thermal mitigation to work on the modem, the following prerequisites must be present:

- A test SIM, i.e., SIM with MCC in range of 1 to12, should *not* be used. A test SIM will enable a GCF test flag, resulting in disabling thermal mitigation.
- Data throttling requires the network/testbox to support dynamic scheduling.

## 4.10.1 Common NV settings

The following sections describe NV items required to be set that are applicable to all radio technologies.

### 4.10.1.1 NV 947 – GPRS-enabled Anite GCF

This NV is required to enable/disable the GCF Testing mode in the device. The NV is set as NV 947: <GCF_flag>.

If GCF Testing mode is enabled, i.e., the GCF flag is set, the thermal mitigation algorithm is disabled. Therefore, it is required to disable the GCF flag by setting NV 947: 0.

Also, note that a test SIM, i.e., SIM with MCC in range of 1 to 12, should *not* be used. This kind of a SIM will enable a GCF test flag automatically, and hence thermal mitigation will be disabled.

### 4.10.1.2 NV 6241 – Thermal monitor timers

NV 6241: <Basic Loop>, <Mitigation Loop>, <t_down>,<misc timer>, <pa change backoff>

It is recommended to set NV 6241: 5000, 5000, 5000, 5000, 1.

## 4.10.2 LTE technology-specific NVs/EFS

### 4.10.2.1 NV 65675 – CFM monitor enable masks

- By default (if the NV item is not written), all monitors including the thermal monitor are enabled (one still needs to configure the DEM thermal threshold, as described in Section 4.1).
- To disable the thermal mitigation for LTE, set the 2 LSBs of the mask to 0, i.e., 0xFFFFFFFC.
- To revert to the default mask, remove the EFS file /nv/item_files/modem/utils/cfm/cfm_monitor_enabled_masks using QPST™/EFS Explorer.
- The only supported values are 0xFFFFFFFC (disable LTE) and 0xFFFFFFFF (enable all monitors in all modes). All other values are not supported and may cause unknown behavior.

#### 4.10.2.2 NV 65611 – LTE FC MAC BSR target data rates

- By default (if the NV item is not written), the default target MAC-level data rate configuration for UL data throttling is as follows:

  Uint8 num_state = 10;

  Uint8 default_state = 5;

  Uint16 reserved = 0; /* keep this set to 0 */

  /* number of bytes per TTI (ms) */

  Uint32 target_rate[0] = 6250 (50 Mbps)

  Uint32 target_rate[1] = 5000 (40 Mbps)

  Uint32 target_rate[2] = 3125 (25 Mbps)

  Uint32 target_rate[3] = 1250 (10 Mbps)

  Uint32 target_rate[4] = 625 (5 Mbps)

  Uint32 target_rate[5] = 125 (1 Mbps)

  Uint32 target_rate[6] = 63 (500 kbps)

  Uint32 target_rate[7] = 13 (100 kbps)

  Uint32 target_rate[8] = 6 (50 kbps)

  Uint32 target_rate[9] = 1 (10 kbps)

- To change the data rate configuration, write to num_state, default_state, target_rate[0]… target_rate[num_state-1]. The data rate is expressed in number of bytes per millisecond.

  To revert to the default configuration, remove the file /nv/item_files/modem/lte/common/ lte_fc_macul_target_rates using QPST/EFS Explorer.

For flow control target data rates (NV 65611), the default value is 10 states, as described above. The default rate is currently set to 1 Mbps. The minimum rate can be set to a lower value, but the recommendation is to set it to a value that would meet the requirement for control channels and delay-sensitive applications. Setting this to 0 is not recommended, as it would shut down the uplink and could cause the call to eventually drop. The default state should be chosen carefully to guarantee a timely response that reduces the temperature. By setting the default state to 5 in the default configuration, the data rate is initially reduced to 1 Mbps after uplink flow control has been triggered.

#### 4.10.2.3 NV 65676 – CFM state remaining time

- By default (if the NV item is not written), the thermal step timer is set to 15 sec.
- The thermal step timer can be changed from 0 to 255 sec by writing to the NV item.
- To revert to the default step timer, remove the EFS file /nv/item_files/modem/utils/cfm/cfm_thermal_step_timer_in_sec using QPST/EFS Explorer.

This timer controls how fast the uplink data rate is increased or decreased (depending on the state). It is currently set to 15 sec based on the observations from test results. This setting should correspond to the time it takes for the temperature sensors to converge to a new temperature value.

#### 4.10.2.4 EFS file for Tx backoff

For LTE, when the threshold for Tx backoff is crossed (NV 6242), the PA power is adjusted per the parameters configured in the EFS file (tx_power_backoff located in /nv/item_files/modem/lte/ML1/).

- P_backoff – Initial value for Tx power backoff in dB
    - Recommended initial value – 5 dB
    - At each step n, the value of power backoff is *n x P_backoff*
    - n is increased by one upon each expiry of step_timer
- T_on – The length of time when the UE removes the limit on MTPL
    - Recommended initial value – 50 ms
- T_off – The length of time when the UE reduces MTPL
    - Recommended initial value – 50 ms
- Step_timer – Time spent in each step (see P_backoff)
    - Recommended initial value – 15 sec
- P_min – The minimum value for MTPL
    - Recommended initial value – 10 dBm

Here is the structure of the EFS file in order:

```
/* Initial backoff */
uint16  p_backoff;
/* Maximum value of the backoff */
uint16  p_backoff_max;
/* Time for non-backed-off value of power */
uint16 t_on;
/* Time for backed off Value of power */
uint16 t_off;
/* Timer for each step of the backoff */
uint32 step_timer;
```

As an example, the following hex content of the file is:

05000C00 32003200 983A0000

- P_backoff – 5 dB (0500 for 5 dB)
- Max_backoff – 12 dBm (0C00 for 13 dB)
- T_on – 50 ms (3200 for 50 ms)
- T_off – 50 ms (3200 for 50 ms)
- Step_timer – 15 sec (983A for 15 sec)

### 4.10.3 WCDMA/HSPA technology-specific NVs/EFS

None

### 4.10.4 1X/DO technology-specific NVs/EFS

None

# 5 Modem Thermal Mitigation Details

**NOTE:** Test results are appended as an example of how the mitigation algorithm can be used, and references for results are given. However, the behavior of the algorithm and settings should be based on the form factor of the device under test.

## 5.1 LTE thermal mitigation functioning

### 5.1.1 Uplink flow control and Tx backoff behavior

Uplink flow control consists of:

- Temperature monitor (currently only PA temperature is used to trigger uplink data throttling)
- Centralized flow control manager
- LTE MAC BSR control
- Tx power backoff

Figure 5-1 shows the uplink flow control.

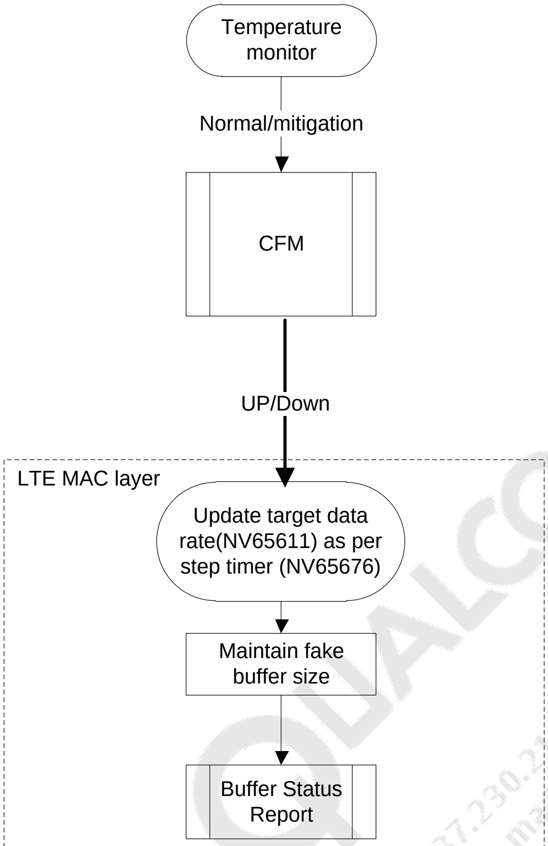


**Figure 5-1 Uplink flow control**

In this section, the uplink throughput in each flow control state is reviewed. The flow control state is derived from the input of one or more thermal sensor subsystems and potentially other resources. Since the focus is on the thermal sensor, only the transitions related to the PA temperature monitor are described.

UL flow control can be in any of four states, Normal, MAC_BRS, Tx Backoff, or Shutdown, as shown in Figure 5-2.

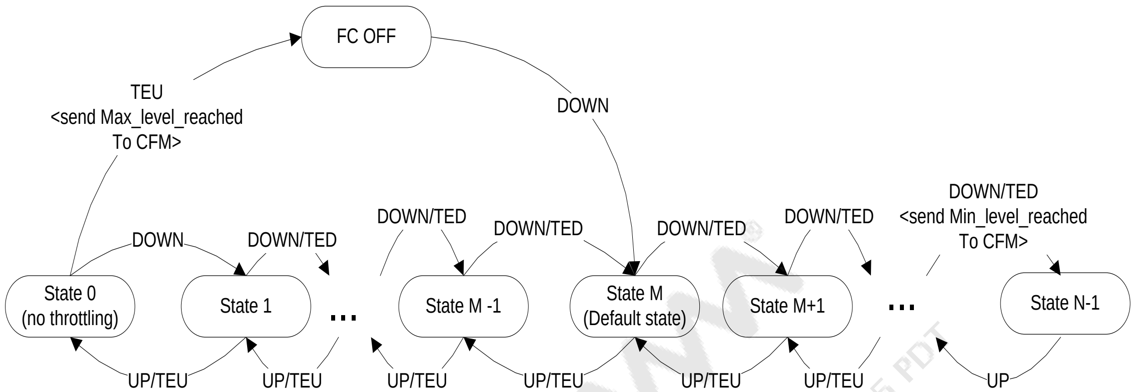


**Figure 5-2 UL flow control**

The UE’s flow control can be categorized in four different states depending on its thermal state, as shown in Figure 5-3 and Figure 5-4.

- Normal state (FC_OFF) – This is the default state of the UE. In this state, the UE is below the configured temperature threshold (PA mitigation threshold), and the UE operates in complete accordance with 3GPP specifications. Once the configured threshold is crossed, the UE moves to the Flow Control (FC_ON) state.

- MAC BSR state – Mitigation 1
  - Upon entering this state from the FC_OFF state, if LTE is up and the MAC_BSR client is registered with the CFM, a DOWN command is sent to the MAC_BSR client.
  - Upon entering this state from the Tx Backoff state, if LTE is up and the Tx_backoff client is registered with the CFM, an UP command is sent to the Tx_backoff client.
  - While in the MAC_BSR state:
    - If a "normal" indication is received from the temperature monitor, and
      - If the MAC_BSR client is registered with the CFM, an UP command is sent to the MAC_BSR client.
      - If the MAC_BSR client is not registered with the CFM, then transition to the FC_OFF state.
    - If a "mitigation" indication is received from the temperature monitor, and if the MAC_BSR client is registered, a DOWN command is sent to the client.
  - If the MAC_BSR client registers with CFM while CFM is in this state, the MAC_BSR client receives a DOWN command immediately after registration.
  - Upon receiving the "Tx backoff" indication from the temperature monitor, transition to the Tx_backoff state.
  - Upon receiving an "emergency" indication from the temperature monitor, transition to the Shutdown state.

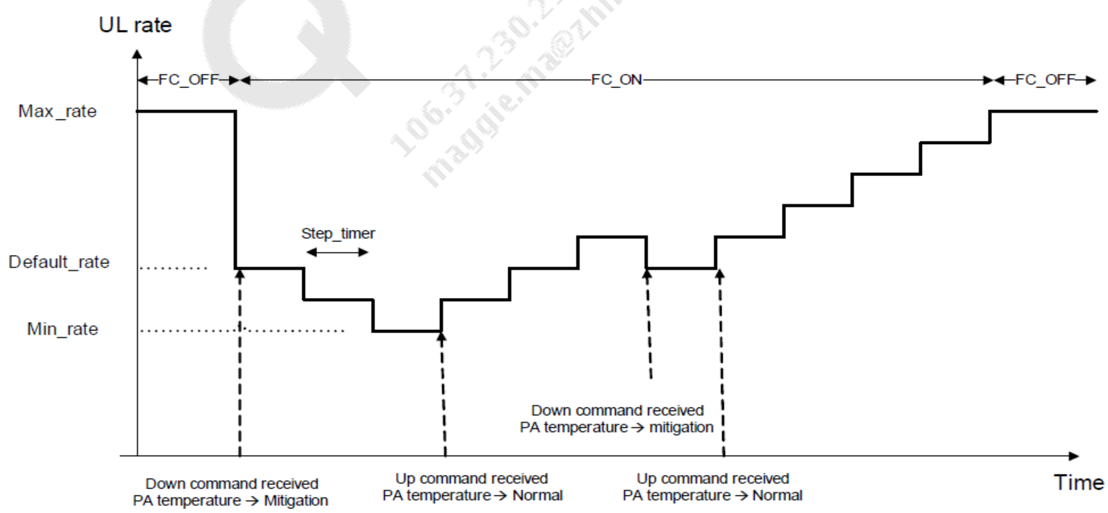


**Figure 5-3 UE states – Mitigation 1**

- Tx Backoff state – Mitigation 2
  - Upon entering this state from the Shutdown state, it is assumed that LTE is not active in the Shutdown state, therefore no action is necessary when transitioning to this state from the Shutdown state.
  - Upon entering this state from the MAC_BSR state, if LTE is up and the Tx_backoff client is registered with CFM, a DOWN command is sent to the Tx_backoff client.
  - While in the Tx_backoff state:
    - If a “mitigation” indication is received from the temperature monitor, and if the Tx_backoff client is registered with CFM, an UP command is sent to the Tx_backoff client; CFM transitions to the MAC_BSR state.
    - If a “normal” indication is received from the temperature monitor, and
      - If the MAC_BSR client is registered, CFM transitions to the MAC_BSR state.
      - If the MAC_BSR client is not registered, CFM transitions to the FC_OFF state.
    - If the MAC_BSR client registers with CFM while CFM is in this state, the MAC_BSR client receives a DOWN command immediately after registration.
    - If the Tx backoff client registers with CFM while CFM is in this state, the Tx_backoff client shall receive a DOWN command immediately after registration.
    - Upon receiving an “emergency” indication from the temperature monitor, transition to the Shutdown state.

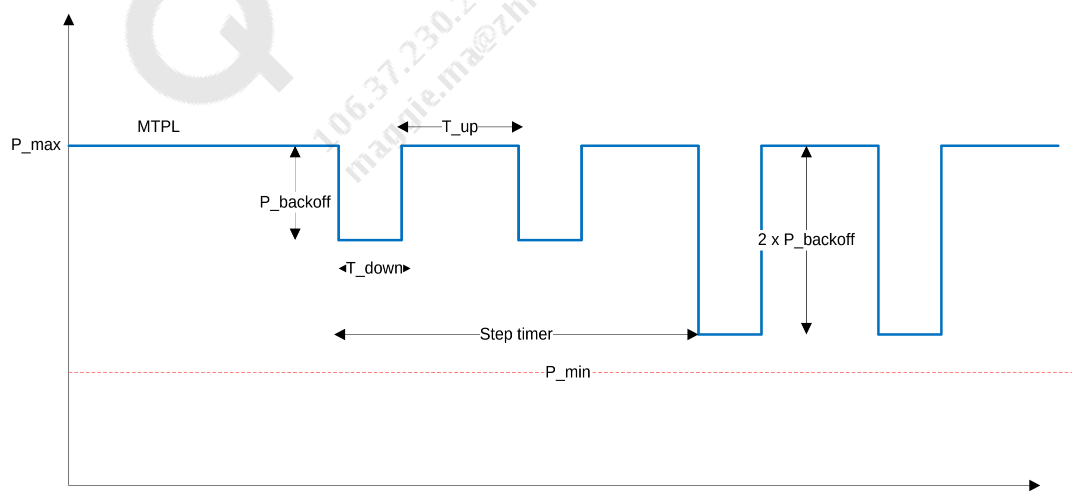


**Figure 5-4 UE states – Mitigation 2**

- Shutdown – Emergency state
  - The behavior in the Emergency state is to drop the RRC connection. The PA is not shut down in this state.

# 5.2 WCDMA/HSPA thermal mitigation

## 5.2.1 RLC data throttling and Tx backoff behavior

The objective of RLC flow control is to reduce power consumption associated with HSPA, as one of the major contributors of power consumption is processing higher data rates.

In the downlink direction, based on the commands from the DEM module, the RLC flow control function will flow control the peer-side RLC (UTRAN) by means of Windows commands when entered into the Mitigation state. A sufficient decrease in the RLC window size, i.e., closing the window, will decrease the amount of data sent to the UE and thereby the processing required. The window size is decreased gradually in steps whenever the periodic timer expires in the Mitigation state. If the device enters the Emergency state, the RLC window size with value 1 is sent to the peer entity in order to go to the lowest possible data throttling. Conversely, if the basic state is entered, the data flow can be increased by increasing the window size (opening the window).

A similar process of window size adjustment is used in the uplink direction. In this case, since the transmitting entity is located on the UE itself, the window size can be directly controlled without the need for sending any commands over the air to the peer-side RLC.

Tx power backoff happens upon entering the Mitigation 2 state:

- Initialize step counter and MPTL backoff value
- Call RF API to reduce the MTPL by an initial backoff value
- Trigger duty-cycle and step timers

Upon exiting the Mitigation 2 state, call the RF API to restore MTPL to a normal value before backoff.

# 5.3 1X/DO thermal mitigation functioning

The thermal sensor for the power amplifier subsystem is integrated with the Congestion Control Manager (CCM). CCM impacts reverse link flow control only in DO/1X mode of operation. The CCM reduces the reverse link data rate so that the heat generated from the PA can be reduced to an acceptable level, so as to maintain the maximum possible data rate in the system.

If PA sensor mitigation is triggered, the reverse link limits the maximum allowed payload size (or packet size) on certain carriers at the reverse link MAC layer (RMAC) in the DO protocol stack. RMAC performs QoS-based packet prioritization. By allowing a certain packet size, QoS may be maintained. If PA sensor critical is triggered on DO, RMAC on DO sets the maximum payload size of all active carriers to 0 except SLP. The SLP carrier is kept at the default maximum payload size.

If PA sensor mitigation/critical is triggered on 1X, the device shall stop requesting for R-SCH (Reverse Supplemental Channel), which will stop the reverse link traffic on 1X. When the PA sensor Normal state is triggered, the device shall resume R-SCH processing.

# 6 Thermal Management Test Results

Updates will be included in subsequent revisions.