Software Thermal Management

80-N9513-1 B

Qualcomm Confidential and Proprietary

QUALCOMM Incorporated
5775 Morehouse Drive
San Diego, CA 92121-1714
U.S.A.

Revision History

Version	Date	Description
A	Mar 2012	Initial release
B	Mar 2012	Added Steps to Tune Modem Mitigation Thresholds section

Contents

Thermal Management Overview

Requirement

- Thermal Management (TM) required since:
  - Market demand for feature-rich, high-performance devices increases thermal susceptibility
  - Multimedia-centric features like SVLTE, SVDO, Mi-Fi, multicore GHz CPUs, graphics processors, HD video increase in thermal dissipation
  - Smaller/thinner designs lead to smaller PCBs, leading to concentrating power density
  - Safety concerns with uncontrolled higher temperature
  - Carrier acceptability touch temperature limits need to be met
  - Component damage possible with uncontrolled higher temperature

Note: Thermal Management/Mitigation and Thermal Management Algorithm/Thermal Mitigation Algorithm are used interchangebly to mean the same thing.

Key Factors Affecting TM

- Key factors affecting TM
  - Form factor/thermal hardware design
  - How long the high performance scenario is run
  - Ambience conditions under which device is tested

Note: Extreme heat-up expected to be rare during ‘typical user’ use case.

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 laboratory, PCI express, and user expectations)
  - Minimize the risk of power-limit constraints
  - Manage the thermal risk and tradeoffs during concurrent operations
  - Allow limited customizable temperature thresholds and methods for power reduction

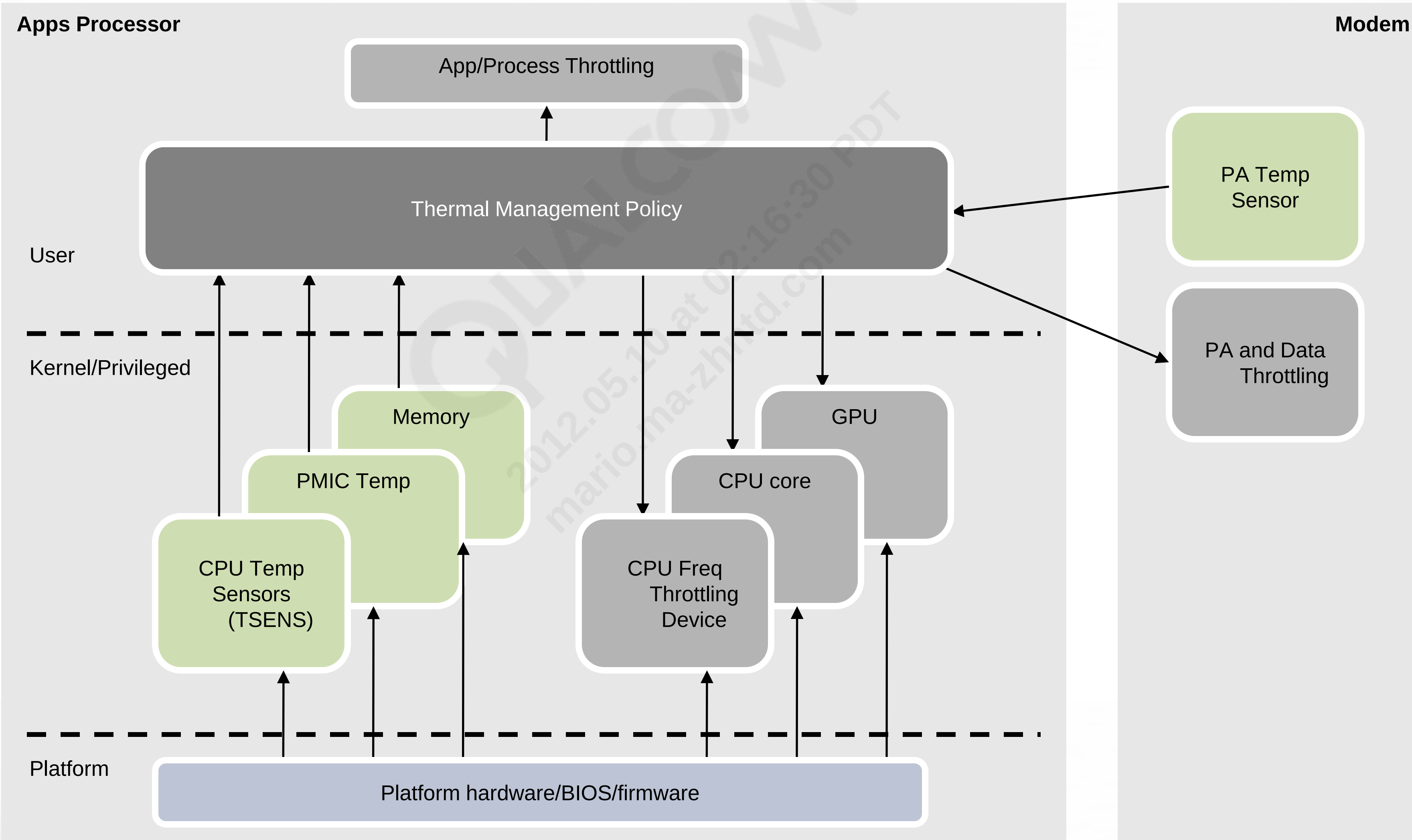
Thermal Management Algorithm

- Thermal Mitigation Algorithm (TMA) allows:
  - Protection against user harm or component damage for rare worst-case conditions
  - Controlling/reducing temperature by trading off device performance
- TMA does *not:*
  - Alter basic power efficiency or heat dissipation properties of the device
  - Change mechanics of the device
  - Fix the cause of heat

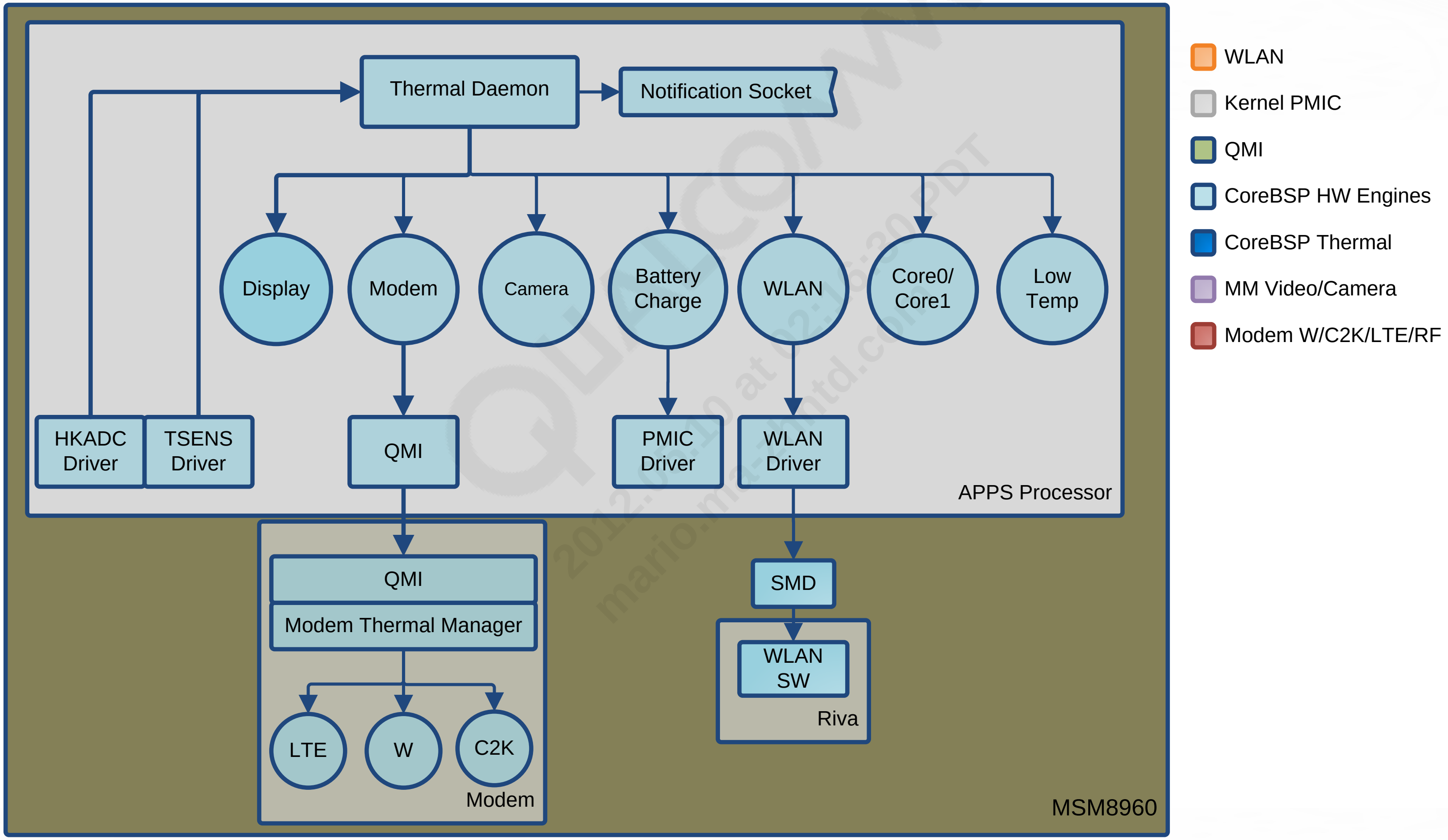
Thermal Management Algorithm Types

- TMA is divided into:
  - Apps side
    - LE-based (for Android™), owned by Qualcomm
    - Windows-based, owned by Microsoft
    - Performed by CPU frequency limiting, battery charge limiting, display intensity controlling
  - Modem side
    - CDMA
    - WCDMA
    - LTE
    - Performed by data flow control, maximum Tx power limiting and call shutdown

Thermal Management Architecture for MSMs

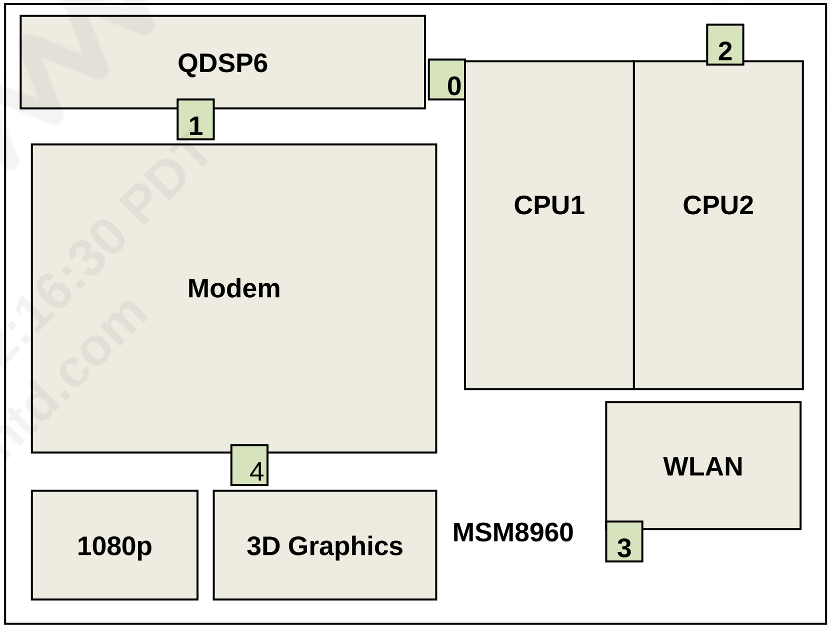


Thermal Daemon Architecture



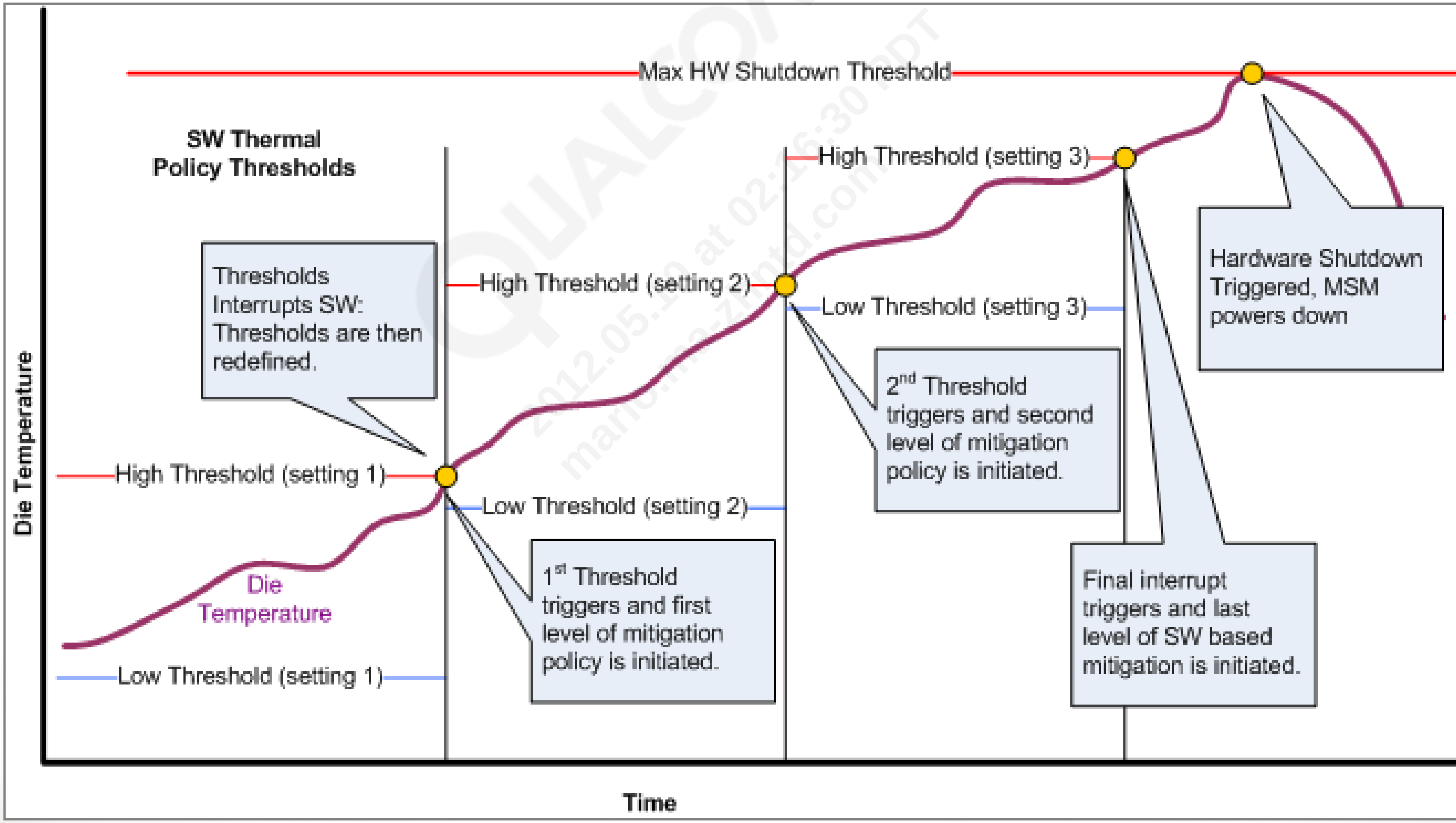
MSM8960 TM Overview

- Sensor-driven algorithm
  - Five sensors on the MSM die (TSENS)
  - External thermistor adjacent to Power Amplifiers (PA) and PMIC
- Device thermal states defined
  - Configurable threshold mapped against states
  - States – No mitigation (normal), mitigation (elevated, severe, and critical)
- 'Cooling Devices'/mitigation devices provide controls to:
  - Achieve thermal control, e.g., CPU (by freq), LCD (by brightness), modem performance (by flow control, Tx backoff, call shutdown)
  - Tradeoff performance for lower power
  - Mitigation device type and intensity mapped to thermal states – Configurable
- Motivation for thermal software
  - Saves components/POP memory from reaching critical temperature
  - Protect device external touch temperature



TM vs Temperature Plot (Example)

- MSM has three to eight software thresholds/mitigation levels and one high hardware shutdown limit



Apps Processor Side TM

AP – High Temperature Use Cases

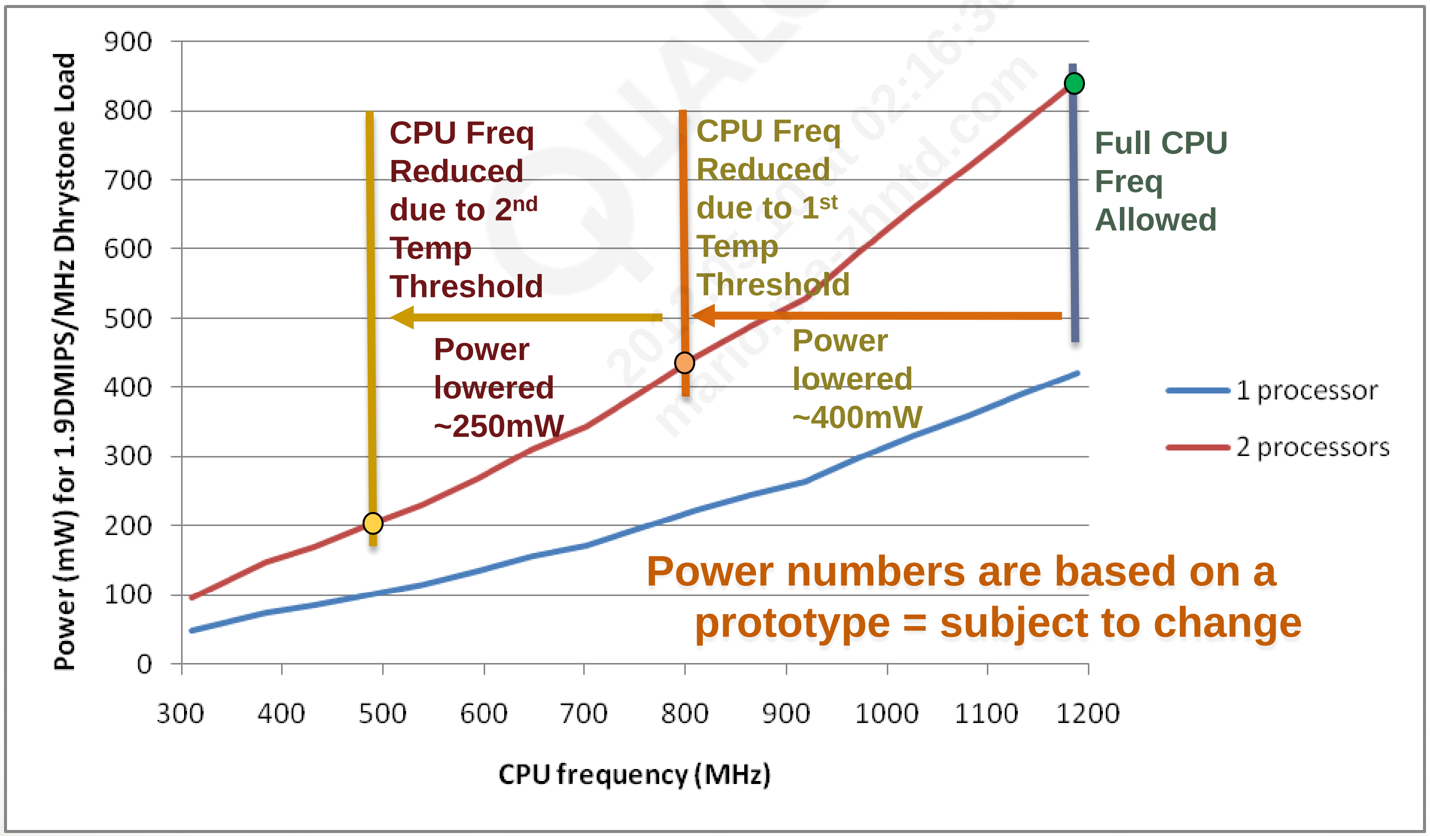
- The following use cases or concurrent combination of these may cause high temperature when performed for prolonged periods:
  - Running powerful/CPU intensive applications with multiple cores at high GHz
  - Running heavy multimedia features with HD videos and 3D graphics
  - Camera operations
  - WLAN operations
  - Device charging, in combination with above

Thermal Management Algorithm

- What TMA does
  - Protects against high temperature scenarios by trading off performance to configured levels
  - Controls temperature of the device by performance tradeoff
  - One of the following is performed to reduce/control temperature
    - Reduce the CPU frequency
    - Reduce LCD brightness
    - Control battery charging

CPU Frequency Scaling Concept

- Graph depicts one potential configuration for two levels of thermal-based throttling, where the maximum CPU frequency is 1188 MHz
  - First threshold reduced maximum CPU capability to ~800 MHz
  - Second threshold reduced maximum CPU capability to ~486 MHz



Note: Power numbers are based on a different MSM; subject to change with MSM type.

Thermal Management Algorithm Implementation

- TM policy is implemented in thermal daemon (thermald)
  - Shared as source code (check ‘drivers\thermal’ in Kernel)
  - Runs in user space on Linux (Android)
  - Starts during boot init phase
  - Needs to be part of init file to get started
  - Monitors temperature reported from:
    - TSENS via TSENS driver
    - Thermistors reported from HKADC driver via QMI interface
  - Takes temperature controlling action based on customizable config file (‘/etc/thermald.conf’ in adb shell)

Thermal Management Actions Breakdown

- Following is an example. It can be configured differently.

	Normal	Elevated	Severe	Critical
CPU throttle	Full frequency range	Highest performance frequencies excluded	Mid to high performance frequencies excluded	Capped at minimum practical frequency
Display intensity	Default intensity	Capped at 90%	Capped at 75%	Capped at 50%
Battery charging backoff	Full charge	Trickle charge	No charge	No charge

Enabling Thermal Management

- For TM to work
  - TSENS must be calibrated
  - *thermald* must be running
  - *thermald.conf* must be present

Requirement	Status	How to check
TSENS calibration	TSENS must be calibrated	In ‘adb shell dmesg’ (android kernel log) search for “tsens_tm_probe: OK”
Process (thermald)	Thermal daemon process should be running on apps side	Check through adb command “ps thermald”
Config file, thermald.conf	Thermal daemon config file should be present on apps side; config file must contain temperature thresholds and actions corresponding to the thresholds	Check through adb command “adb pull /etc/thermald.conf”

Debugging/Checking Thermal Management

- To debug thermal management, thermald must be started in Debug mode.

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

  - Or add ‘debug’ as first line in *thermald.conf* and restart the device.
- The following command will output the thermald log information in the command window.

```
adb logcat -s ThermalDaemon
```

Debugging/Checking Thermal Management (cont.)

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

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

- *grep* ‘Thermal’ in logs can show the important messages.
- Note that temperature will only show in logs once the first thermal threshold of TSENS0 is hit, e.g., for the sample file provided, unless tsens_0 reaches 65°, temperature will not be reported in the log file.

```
01-02 00:00:53.676   948   954 E ThermalDaemon: Sensor[pa_therm0] Temperature : 22.0
01-02 00:00:53.696   948   955 E ThermalDaemon: Sensor[pa_therm1] Temperature : -30.0
01-02 00:00:53.876   948   959 E ThermalDaemon: Sensor[tsens_tz_sensor3] Temperature : 33.0
01-02 00:00:53.876   948   956 E ThermalDaemon: Sensor[tsens_tz_sensor0] Temperature : 33.0
01-02 00:00:53.876   948   958 E ThermalDaemon: Sensor[tsens_tz_sensor2] Temperature : 33.0
01-02 00:00:53.876   948   960 E ThermalDaemon: Sensor[tsens_tz_sensor4] Temperature : 35.0
01-02 00:00:53.876   948   957 E ThermalDaemon: Sensor[tsens_tz_sensor1] Temperature : 31.0
```

To read Temperature from TSENS/Thermistors

- Use following to read the temperatures
  - /sys/devices/virtual/thermal/thermal_zone0/temp – For TSENS0
  - /sys/devices/virtual/thermal/thermal_zone1/temp – For TSENS1
  - /sys/devices/virtual/thermal/thermal_zone2/temp – For TSENS2
  - /sys/devices/virtual/thermal/thermal_zone3/temp – For TSENS3
  - /sys/devices/virtual/thermal/thermal_zone0/temp – For TSENS4
  - /sys/devices/platform/msm_ssbi.0/pm8921-core/pm8xxx-adc /pa_therm0 – For PA Therm 0
  - /sys/devices/platform/msm_ssbi.0/pm8921-core/pm8xxx-adc /pa_therm1 – For PA Therm 1

Configuration File for Thermal Management

- Configuration file
  - Located in /etc/thermald.conf (in adb shell)
  - Allows temperature thresholds
  - Allows cooling devices (CPU, LCD, modem, etc.) and corresponding action per threshold
  - Must contain section for at least TSENS0
  - May contain pa_therm0 only if PA thermistor#1 is present
  - May contain pa_therm1 section only if PA thermistor#2 is present
  - May contain ‘debug’ as first line if thermald needs to be started in Debug mode on startup

Configuration File for Thermal Management (cont.)

- Configuration file format

```
sampling         1000                          : Default sampling period of 1000msec for internal sensor if not otherwise defined
[tsens_tz_sensor0]                             : Sensor 0 (Main sensor) temperature sensor specified for following parameters
sampling         5000                          : Default sampling period of 5000msec
thresholds       55     65      75             : Thresholds defined at temperature of 55, 65 and 75 C
thresholds_clr   52     62      72             : Clearing thresholds of 52, 62, and 72C
actions          cpu   cpu+lcd   cpu+lcd       : Actions are for CPU frequency  and LCD brightness
action_info     1026000 756000+100 384000+50   : Defines new max CPU frequency and LCD brightness with each level
```

- Threshold action is reverted if temperature falls below clearing temperature
- Specific form factor should be characterized through experiments to decide values of temperature thresholds and actions (see appendix)

Sample FFA/MTP thermald.conf File (After Tuning)

```
debug
sampling          5000

[pa_therm0]
sampling          1000
thresholds        70    80     90
thresholds_clr    65    75     85
actions           modem    modem    modem
action_info       1    2    3

[tsens_tz_sensor0]
sampling          1000
thresholds        65      90      93      96      99      102     105
thresholds_clr    62      87      90      93      96      99      102
actions           cpu     cpu     cpu     cpu     cpu     cpu     shutdown
action_info       1512000 1296000 1188000 918000  756000  648000  5000

[tsens_tz_sensor1]
sampling          1000
thresholds        75
thresholds_clr    72
actions           none
action_info       0
```

Sample FFA/MTP thermald.conf File (After Tuning) (cont.)

```
[tsens_tz_sensor2]
sampling         1000
thresholds       75
thresholds_clr   72
actions          none
action_info      0

[tsens_tz_sensor3]
sampling         1000
thresholds       75      78      81      84      87      90
thresholds_clr   72      75      78      81      84      87
actions          cpu     cpu     cpu     cpu     cpu     shutdown
action_info      1296000 1188000 918000  756000  648000  5000

[tsens_tz_sensor4]
sampling         1000
thresholds       75
thresholds_clr   72
actions          none
action_info      0
```

Modem Side TM

Modem – High Temperature Use Cases

- The following use cases or concurrent combination of these may cause high temperature when performed for prolonged periods:
  - Performing throughput test (particularly UL)
  - Date throughput under poor RF conditions where Tx power is very high, while UE is transmitting continuously
  - High DL data rates in conjunction with applications such as video

Thermal Management Algorithm

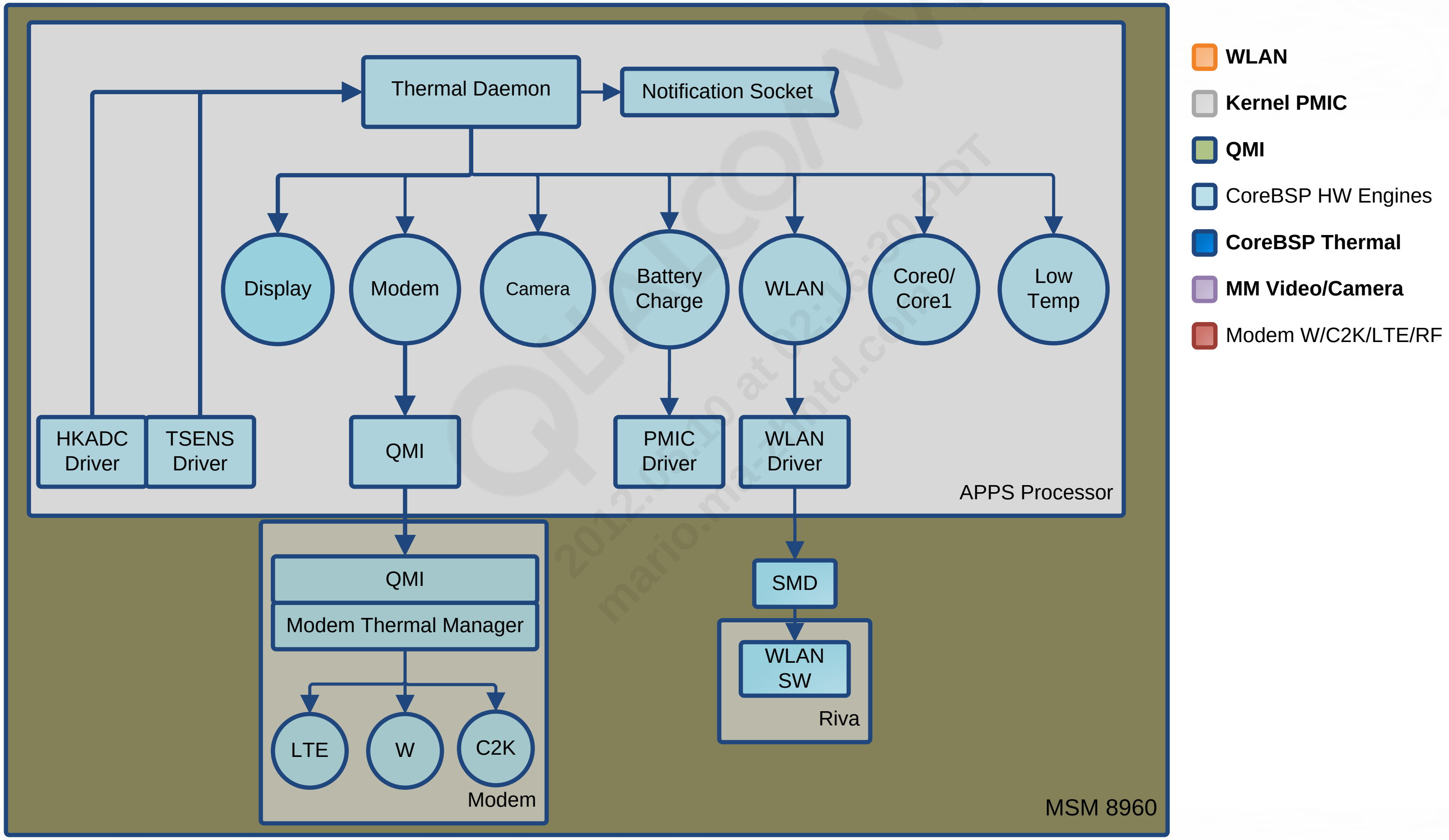
- What thermal management algorithm does
  - Protects against high temperature scenarios by trading off performance to configured levels
  - Controls temperature of the device by performance tradeoff
  - One of the following is performed to reduce/control temperature
    - Reduce the UL/DL throughput rate by duty-cycling
    - MTPL backoff
    - Call shutdown/limited service (only E911 allowed)

Note: Thermal management on the modem is only active when using a non-GCF SIM, i.e., a SIM that is not programmed with MCC-MNC (1-1)).

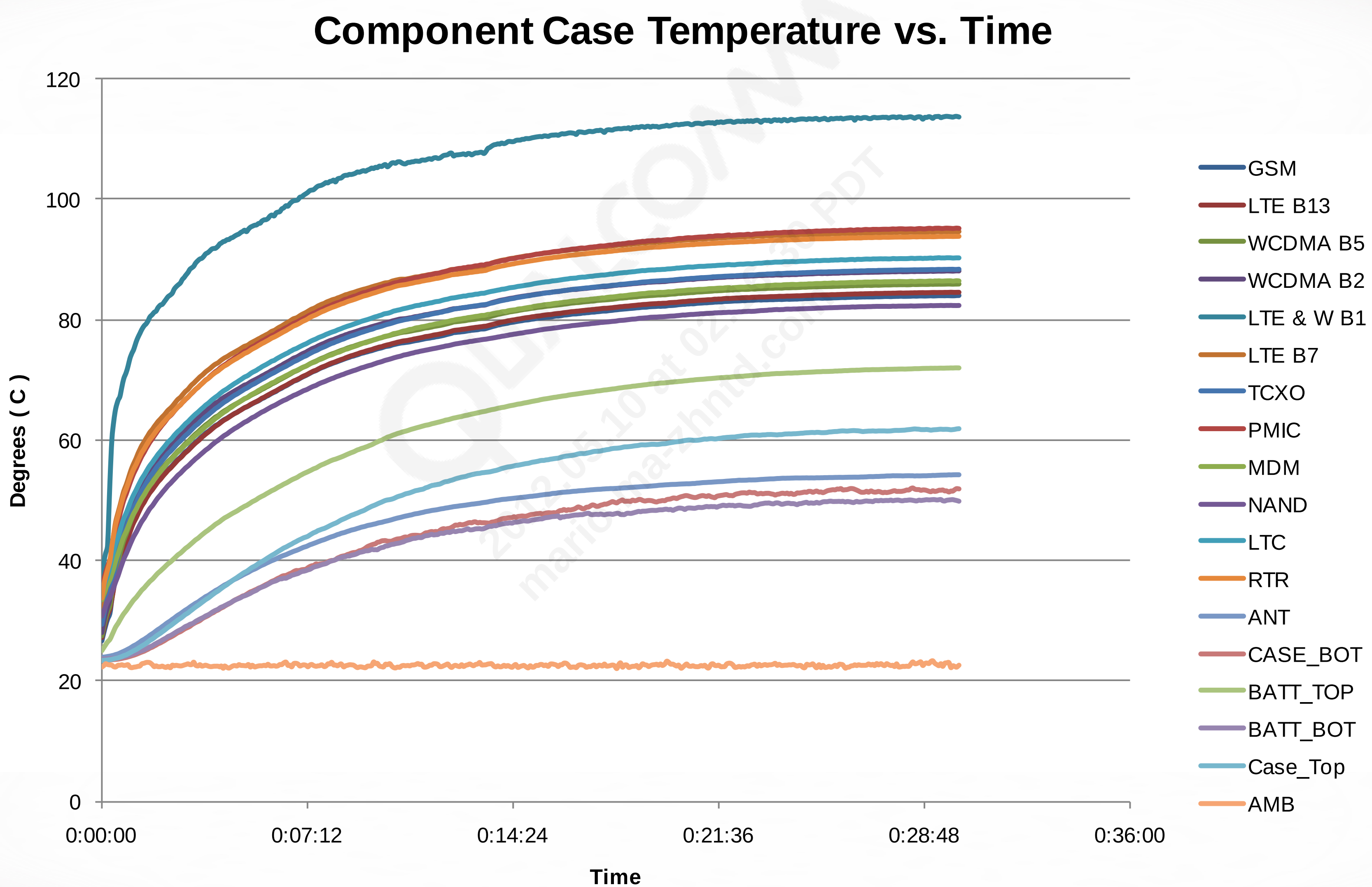
Thermal Management Architecture

- Thermal Management Architecture
  - *Thermald* – Receives temperature reading from thermistors and TSENS
  - PA – Typically, the hottest component in modem centric scenarios
  - *Thermald* – Based on temperature and configured thresholds, sends command to modem
  - Individual active RAT receives the command and mitigates by
    - Reducing the UL/DL throughput rate
    - MTPL backoff
    - Calling shutdown/limited service (only E911 allowed)

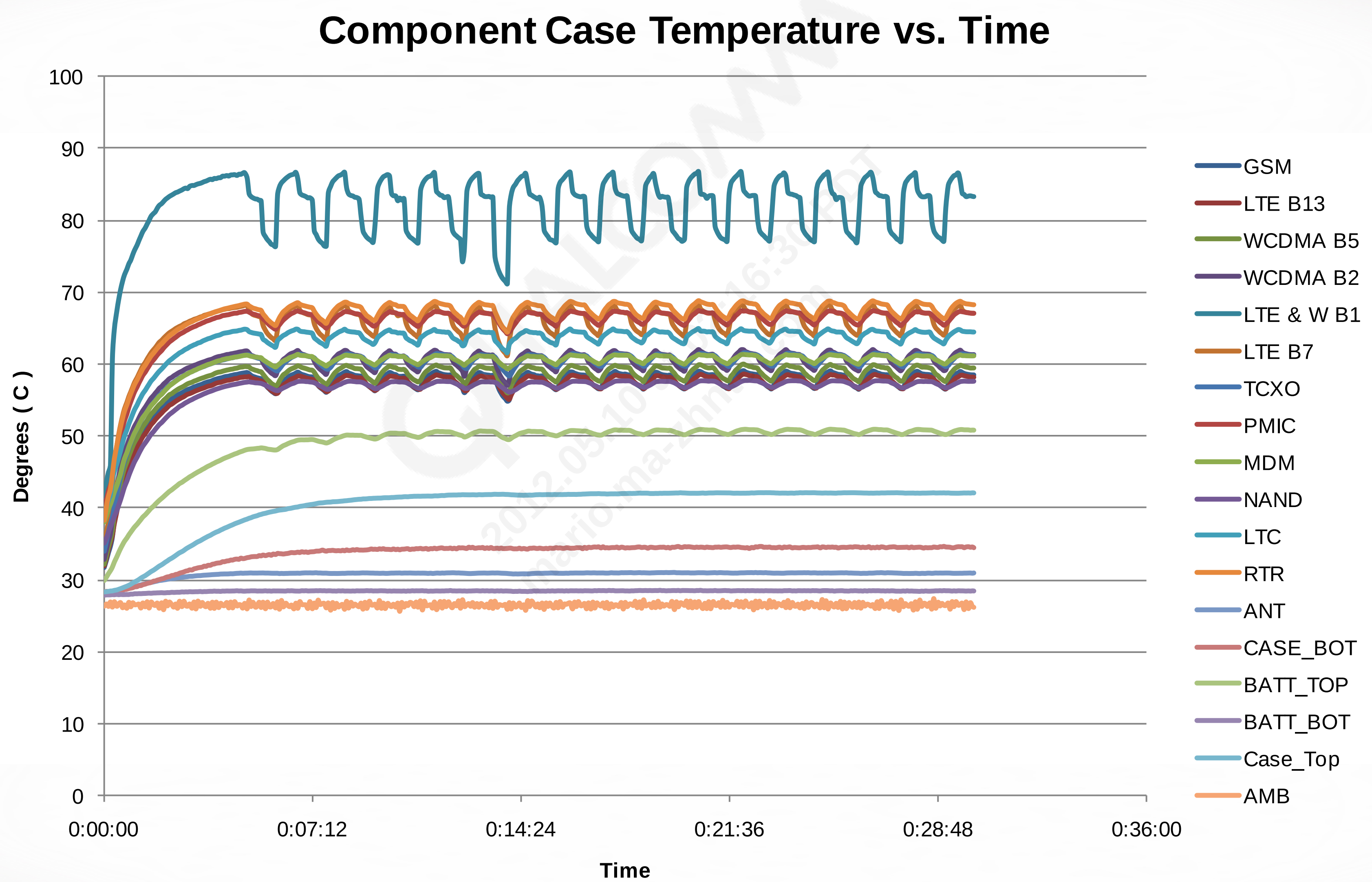
Thermal Daemon Architecture



Modem Thermal Graph without TM



Modem Thermals with TM



Modem – Thermal Management Implementation

- Managed by thermald
- Temperature reported to thermald by HKADC via QMI interface
- Actions/thresholds defined in the thermal daemon configuration file (thermald.conf)
  - One of the following is performed to reduce/control temperature
    - Limit the uplink data throughput while keeping the original power class, while doing duty-cycling/DTX; this is preferred/first method
    - Reduce the power class of the device, lowering the maximum Tx power to limit the power dissipation of the power amplifier; this is not the preferred method; however, it is desirable as another tool to reduce the probability that the emergency state is reached
    - Emergency state – Call shut down; device goes to limited service and allows only E911 calls

Configuration File for Thermal Management

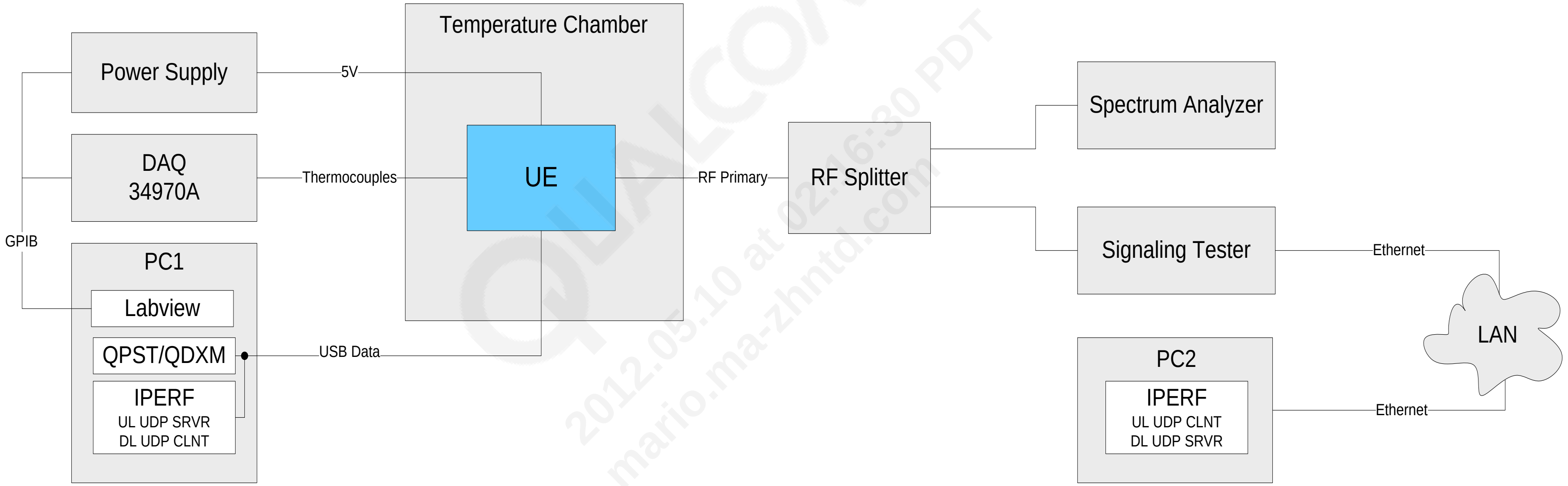
- Config file – thermald.conf located in */etc/thermald.conf* (in adb shell)
- See slide “Configuration File for Thermal Management” for details

Configuration File Example for Modem Management

- Three levels for modem management can be configured in thermald.conf
  - 0 – No mitigation (default)
  - 1 – Data throttling
  - 2 – Tx power backoff
  - 3 – Call drop/maximum mitigation/only E911 calls allowed
- Modem configuration in thermald.conf

```
[pa_therm0]
sampling         1000
thresholds       70      80       90
thresholds_clr   65      75       85
actions          modem     modem     modem
action_info      1         2         3
```


Test Setup



Temperature Chamber
Power Supply
5V
DAQ
34970A
Thermocouples
UE
RF Primary
RF Splitter
Spectrum Analyzer
Signaling Tester
Ethernet
GPIB
PC1
Labview
QPST/QDXM
USB Data
IPERF
UL UDP SRVR
DL UDP CLNT
LAN
PC2
IPERF
UL UDP CLNT
DL UDP SRVR
Ethernet

Test Setup Requirement

- For modem thermal management, following are test setup requirements:
  - Thermal management is only active when using a non-GCF SIM, i.e., a SIM that is not programmed with MCC-MNC (1-1)).
  - For WCDMA flow control to work, network/network simulator *must* support WIN_SIZE SUFI
  - For LTE flow control to work, network/network simulator *must* support dynamic scheduling (i.e., scheduling based on buffer status reported by UE).
    - This has been verified to work on following test boxes:
      - Anritsu 8430 v2.10b + RTD 4.8.0
      - Aeroflex

Note: Contact test box vendor to get support for dynamic scheduling.

LTE TM

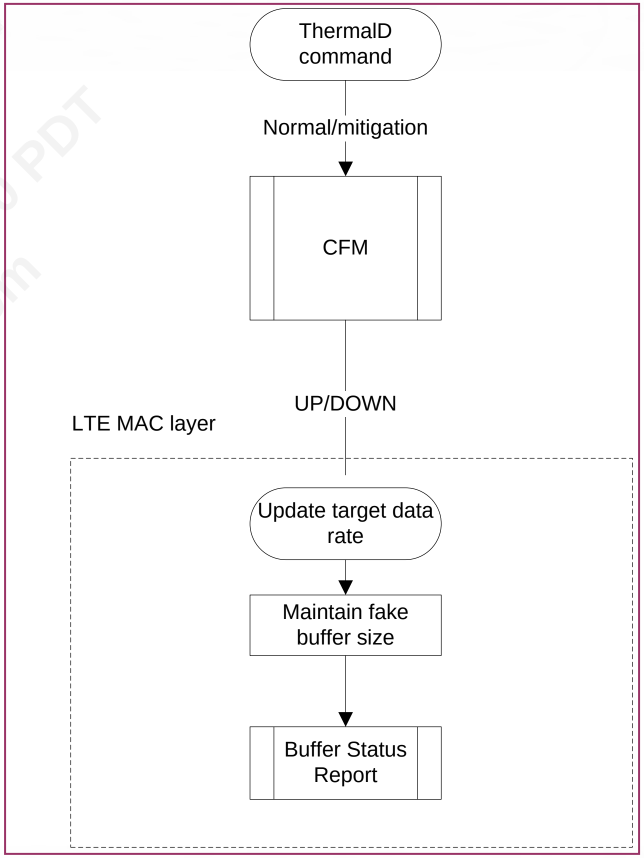
Thermal Management Actions Breakdown

- Following is an example, which can be configured differently.

	Normal	Elevated	Severe	Critical
Modem/data throttle (only UL)	No restrictions	Start throttling upload data	Continue to throttle upload	Disable all data modes
Maximum Tx power throttling	Full power range	Full power range	Limit maximum Tx power	N/A (data modes already disabled)
Emergency/call shutdown	None	None	None	Call released/limited service; only E911 allowed

Mitigation Level 1 – UL Data Throttling

- NV 65611 defines the flow-control target LTE data rates; these data rates are expressed in number of bytes per millisecond
- NV 65676 step timer in seconds for changing rate states (default 15 sec)
- With the centralized flow manager, UE will send fake buffer status reports to the network based on the target rate; therefore, the network will assign lower grant based on same



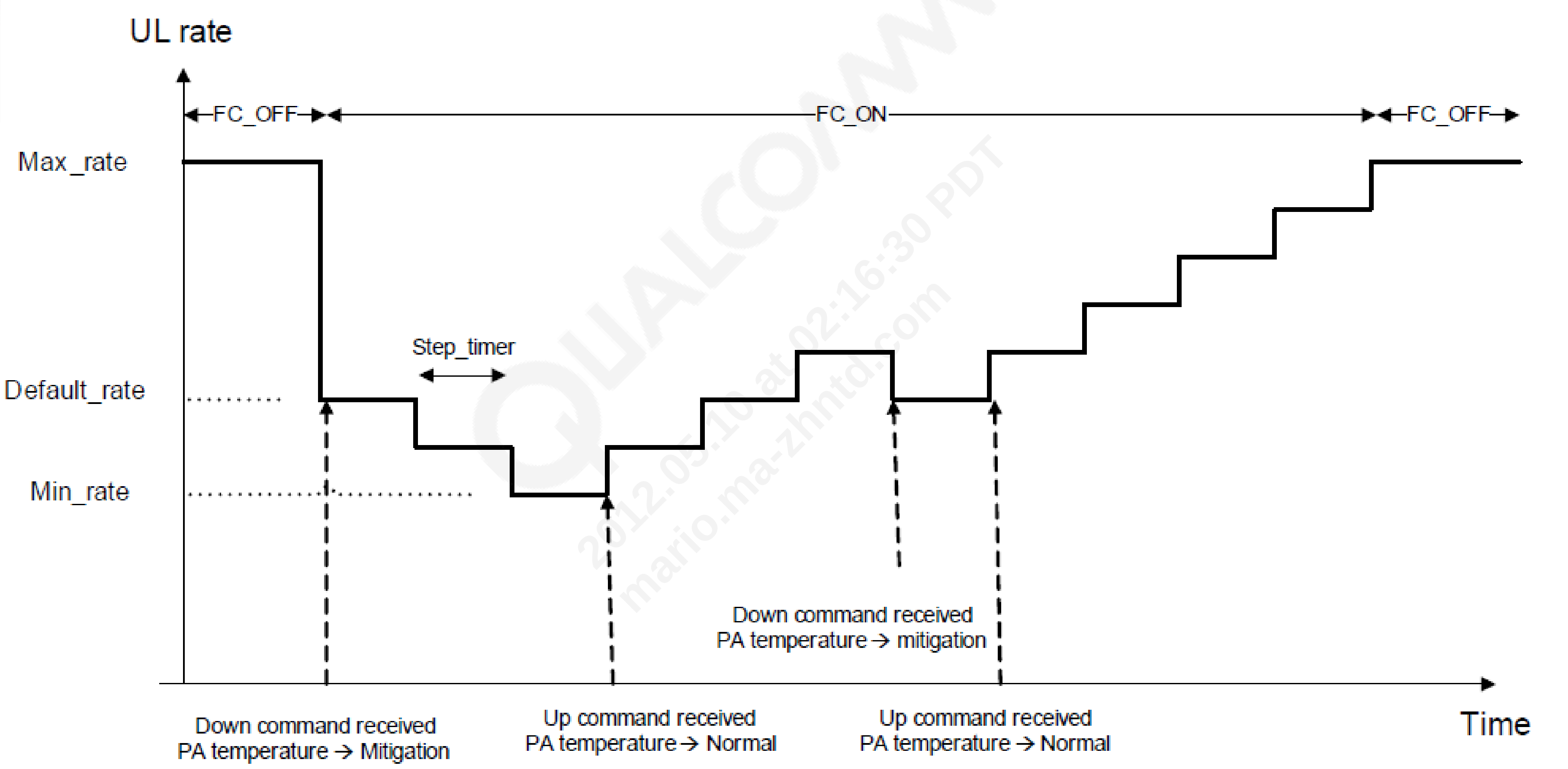
Mitigation Level 1 – UL Data Throttling (cont.)

- For modem mitigation level 1, NV 65611 and NV 65676 are needed to be configured as shown

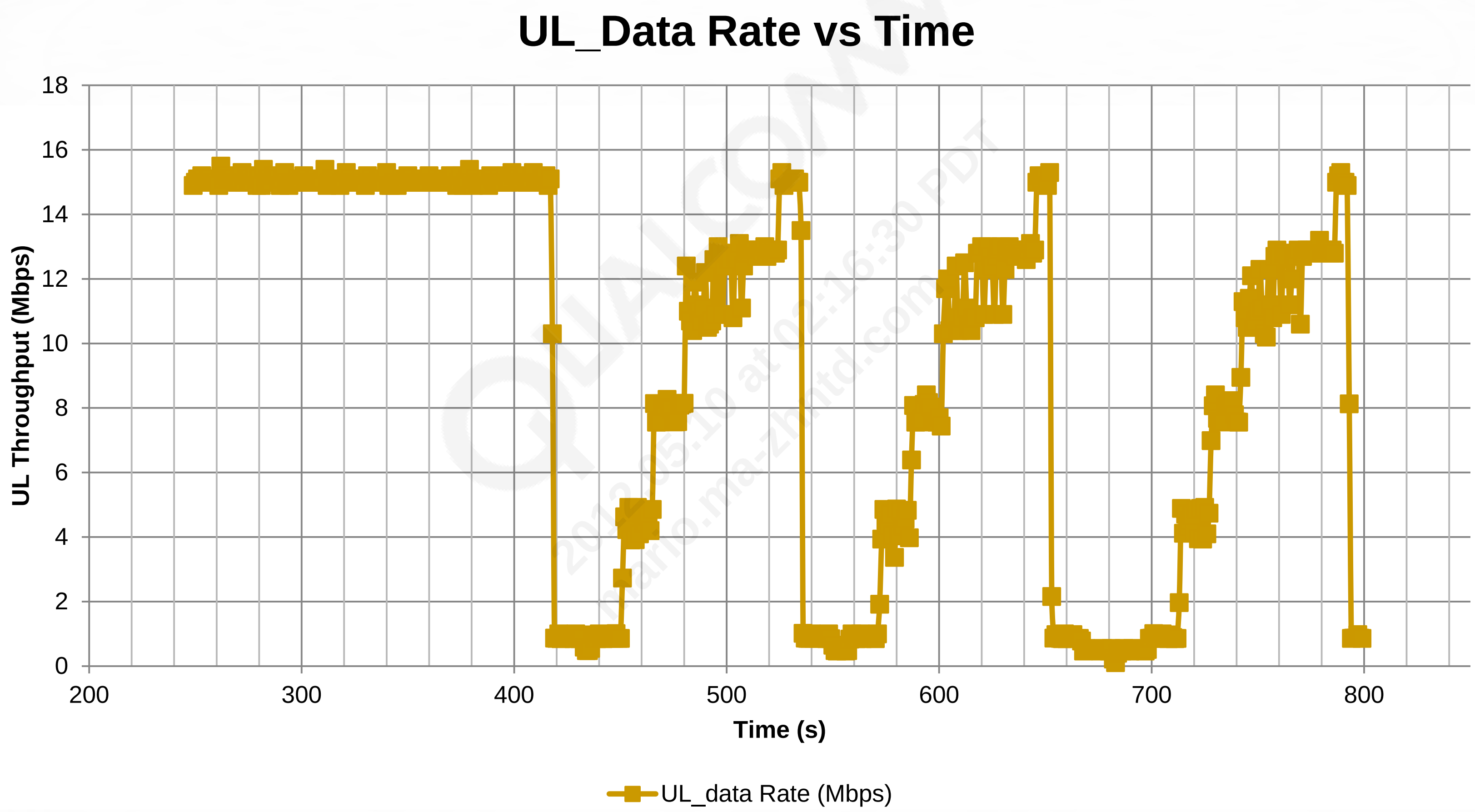
NV item	NV item description	Configuration for TM
NV 65611	Flow-control target LTE data rates; data rate is expressed in number of bytes per millisecond	▪ uint8 num_state = 10; ▪ uint8 default_state = 5; ▪ uint16 reserved = 0; ▪ target_rate[0] = 6250 ▪ target_rate[1] = 5000 ▪ target_rate[2] = 3125 ▪ target_rate[3] = 1250 ▪ target_rate[4] = 625 ▪ target_rate[5] = 125 ▪ target_rate[6] = 62 ▪ target_rate[7] = 12 ▪ target_rate[8] = 6 ▪ target_rate[9] = 1
NV 65676	Step timer in seconds for changing rate states	15 sec

Note: Thermal management is only active when using a non-GCF SIM, i.e., a SIM that is not programmed with MCC-MNC (1-1)).
For LTE flow control to work, network/network simulator *must* support dynamic scheduling (i.e., scheduling based on buffer status reported by UE.

Mitigation Level 1 – UL Data Throttling (cont.)

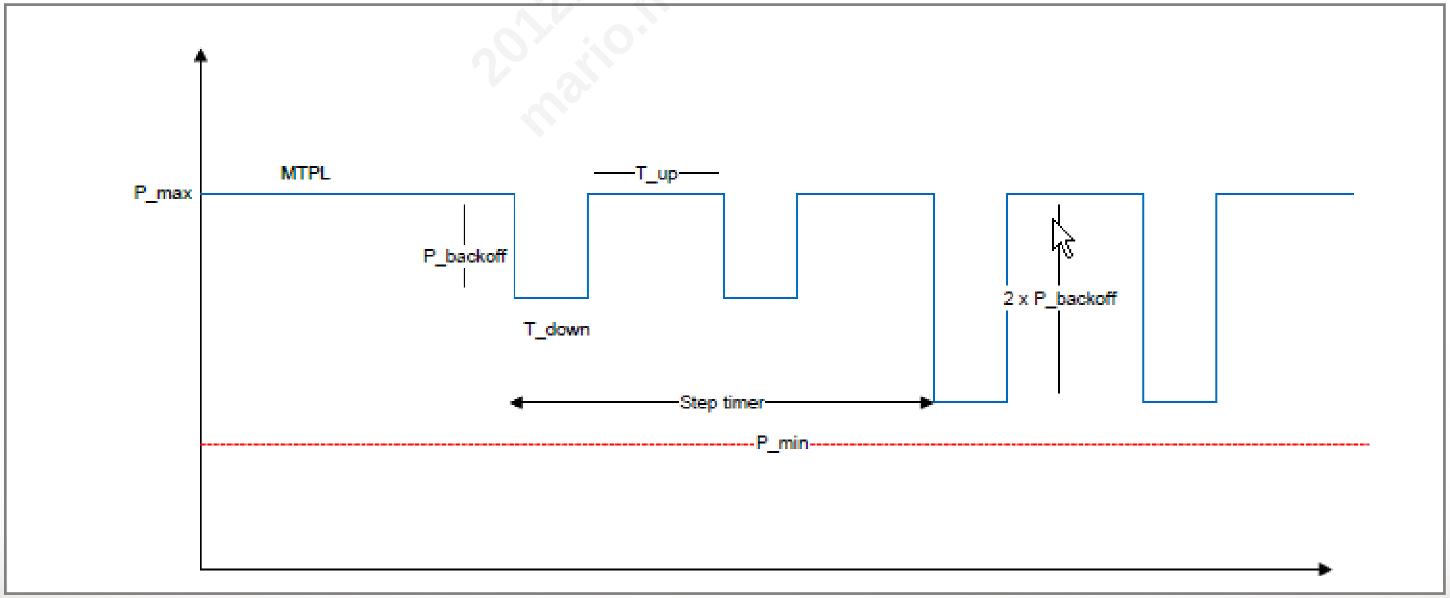


Modem Mitigation Level 1 – UL Data Throttling Example



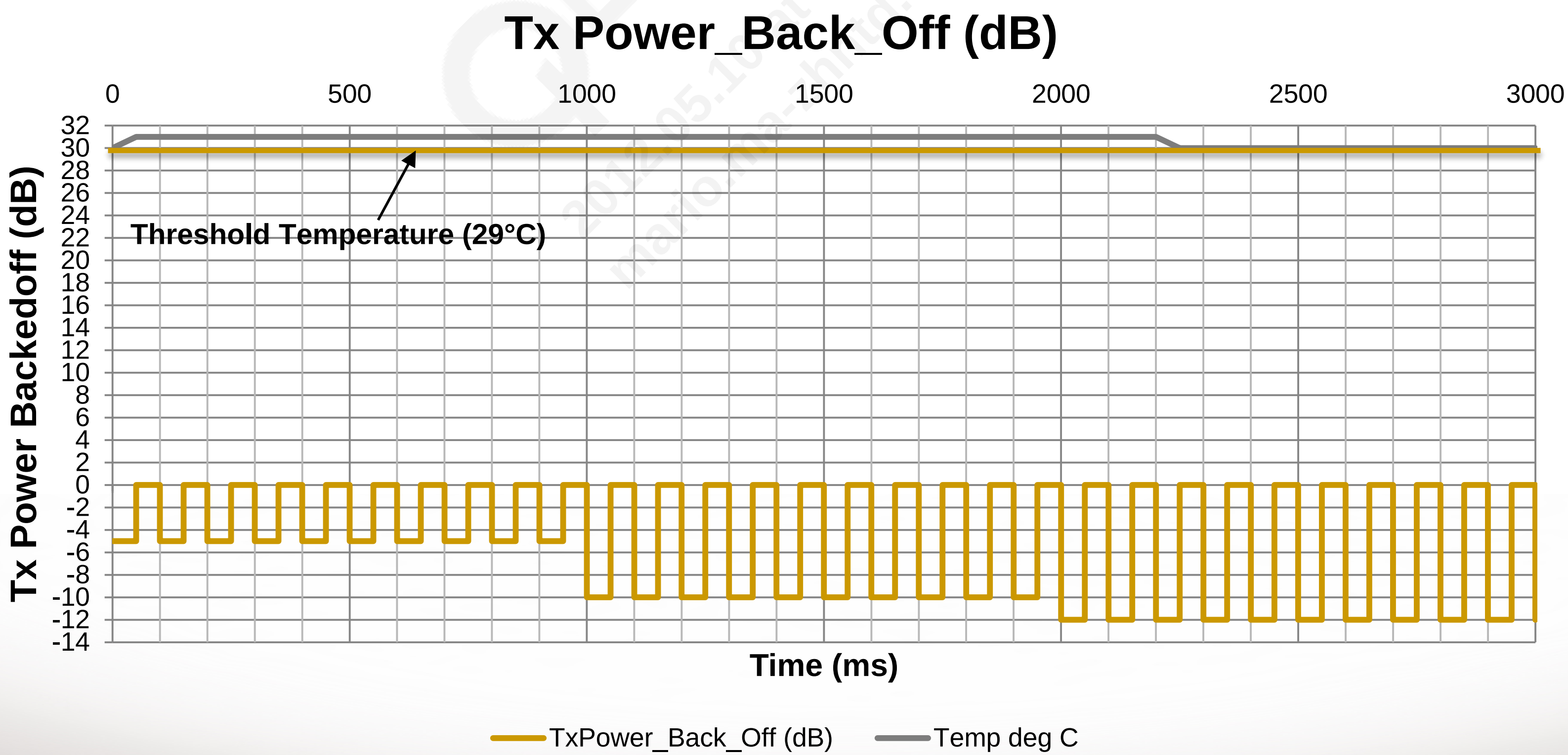
Mitigation Level 2 – Tx Power Backoff

- For LTE in the Mitigation Level 2 state, the PA power will be adjusted as per the parameters configured in the EFS file (tx_power_backoff located in "/nv/item_files/modem/lte/ML1/").
- Some of the values that can be configured in the EFS file are:
  - P_backoff – Initial value for Tx power backoff in dB (at each step n, the value of power backoff is n x P_backoff)
  - T_on – The length of time when the UE removes the limit on MTPL
  - T_off – The length of time when the UE reduces MTPL
  - Step_timer – Time spent in each step (see P_backoff)



Mitigation Level 2 – Tx Power Backoff (cont.)

- Tx power Backoff EFS File parameters
  - P_backoff – 5 dB
  - T_on – 50 ms
  - T_off – 50 ms
  - Step_timer – 1 sec
  - Max_backoff – 12 dBm



Mitigation Level 2 – Tx Power Backoff (cont.)

- Structure of tx_power_backoff located in /nv/item_files/modem/lte/ML1/

```
/* 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;
```

- Example
  - If hex content of the file is:
    - 05000C00 32003200 983A0000 then
      - 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)

Mitigation Level 3 – Emergency Shutdown

- When this state is entered:
  - RRC connection is released
  - Device camps in limited service
  - Allows only E911 call until the thermal mitigation level is reduced

Debugging: Log Packets

- The following log packets/F3 are required for debugging LTE thermal-related issues:
  - Log packet [0xB146] LTE LL1 AGC Tx report – To monitor Tx power
  - Log packet [0xB064] MAC UL transport block – To monitor UL flow control
  - Log packet [0x14D8] temperature monitor
  - MSG F3 [00043/02] flow controller

Debugging (cont.)

```
1980 Jan  6  00:02:36.644 [00]          0x14D8 Temperature Monitor Log
-----------------------------
|   |    |Sensor|Temperature|
|#  |SSID|ID    |Reading    |
-----------------------------
|  0|   0|     0|         74|
|  1|   2|     0|         39|

1980 Jan  6  00:02:41.651 [00]          0x14D8 Temperature Monitor Log
-----------------------------
|   |    |Sensor|Temperature|
|#  |SSID|ID    |Reading    |
-----------------------------
|  0|   0|     0|         75|
|  1|   2|     0|         39|

1980 Jan  6  00:02:46.654 [00]          0x1FEB Extended Debug Message
   cfm_monitor.c          985           H             CFM proc monitor state change monitor=0, cmd=7, step_timer=15000
1980 Jan  6  00:02:46.654 [00]          0x1FEB Extended Debug Message
   cfm_client.c           415           H             CFM issued FC command to client=0, cmd=7, step_timer=15000
```

Debugging – Entering Flow Control State

```
1980 Jan  6  00:02:46.659 [00]        0x14D8 Temperature Monitor Log
-----------------------------
|   |    |Sensor|Temperature|
|#  |SSID|ID    |Reading    |
-----------------------------
|  0|   0|     0|         77|
|  1|   2|     0|         40|

1980 Jan  6  00:02:46.659 [00]        0x1FEB Extended Debug Message
   demmqdsp_therm.c       1303        H            DEMMQDSP: Exit Basic, Enter Mitigation State

1980 Jan  6  00:02:51.662 [00]        0x1FEB Extended Debug Message
   cfm_monitor.c          985         H            CFM proc monitor state change monitor=0, cmd=1, step_timer=15000

1980 Jan  6  00:02:51.662 [00]        0x1FEB Extended Debug Message
   cfm_client.c           415         H            CFM issued FC command to client=0, cmd=1, step_timer=15000

1980 Jan  6  00:02:51.666 [00]        0x14D8 Temperature Monitor Log
-----------------------------
|   |    |Sensor|Temperature|
|#  |SSID|ID    |Reading    |
-----------------------------
|  0|   0|     0|         72|
|  1|   2|     0|         40|
1980 Jan  6  00:02:51.667 [00]        0x1FEB Extended Debug Message
   demmqdsp_therm.c       1368        H            DEMMQDSP: Exit Mitigation, Enter Basic State
```

Debugging – Entering Emergency State

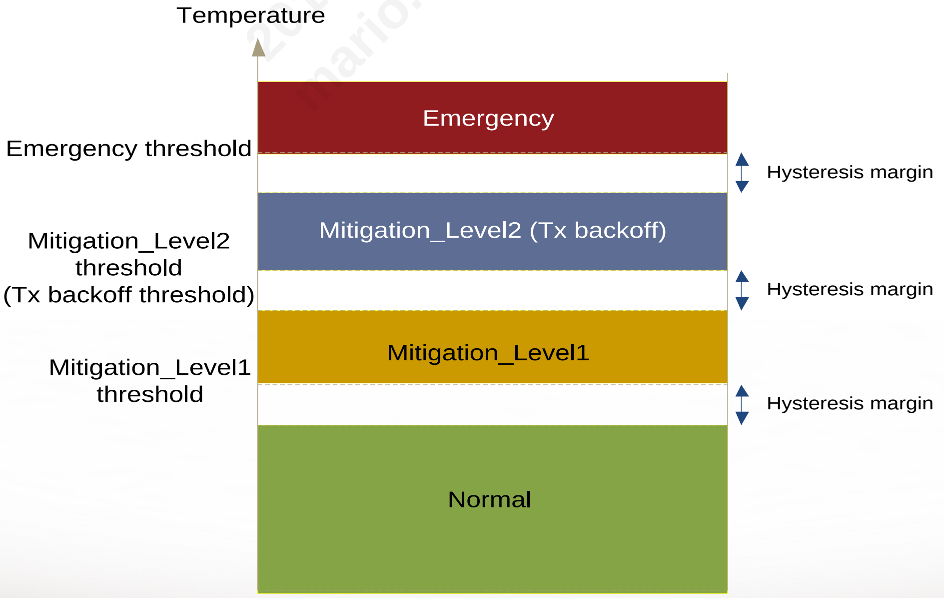
```
1980 Jan  6  00:25:11.926 [00]         0x14D8 Temperature Monitor Log
-----------------------------
|   |    |Sensor|Temperature|
|#  |SSID|ID    |Reading    |
-----------------------------
|  0|   0|     0|         89|
|  1|   2|     0|         67|

1980 Jan  6  00:25:26.944 [00]         0x1FEB Extended Debug Message
   cfm_monitor.c          985          H            CFM proc monitor state change monitor=0, cmd=31, step_timer=15000
1980 Jan  6  00:25:26.944 [00]         0x1FEB Extended Debug Message
   cfm_client.c           415          H            CFM issued FC command to client=0, cmd=31, step_timer=-1
1980 Jan  6  00:25:26.944 [00]         0x1FEB Extended Debug Message
   lte_rrc_crp.c          1523         H            CRP: Processing Initiate Conn Rel indication
1980 Jan  6  00:25:26.945 [00]         0x1FEB Extended Debug Message
   lte_rrc_controller.c   3332         H            RRCC: Received Connection Release Started
1980 Jan  6  00:25:26.945 [00]         0x1FEB Extended Debug Message
   lte_rrc_controller.c   3343         H            RRCC: State is not suspended, moving to CLOSING
1980 Jan  6  00:25:26.945 [00]         0x1FEB Extended Debug Message
   lte_rrc_controller.c   3672         H            RRCC: Transitioned from state CONNECTED to CLOSING

1980 Jan  6  00:25:32.169 [00]         0x14D8 Temperature Monitor Log
-----------------------------
|   |    |Sensor|Temperature|
|#  |SSID|ID    |Reading    |
-----------------------------
|  0|   0|     0|         87|
|  1|   2|     0|         68|
```


CDMA TM

Thermal Management Actions Breakdown

- Following is an example, which can be configured differently.

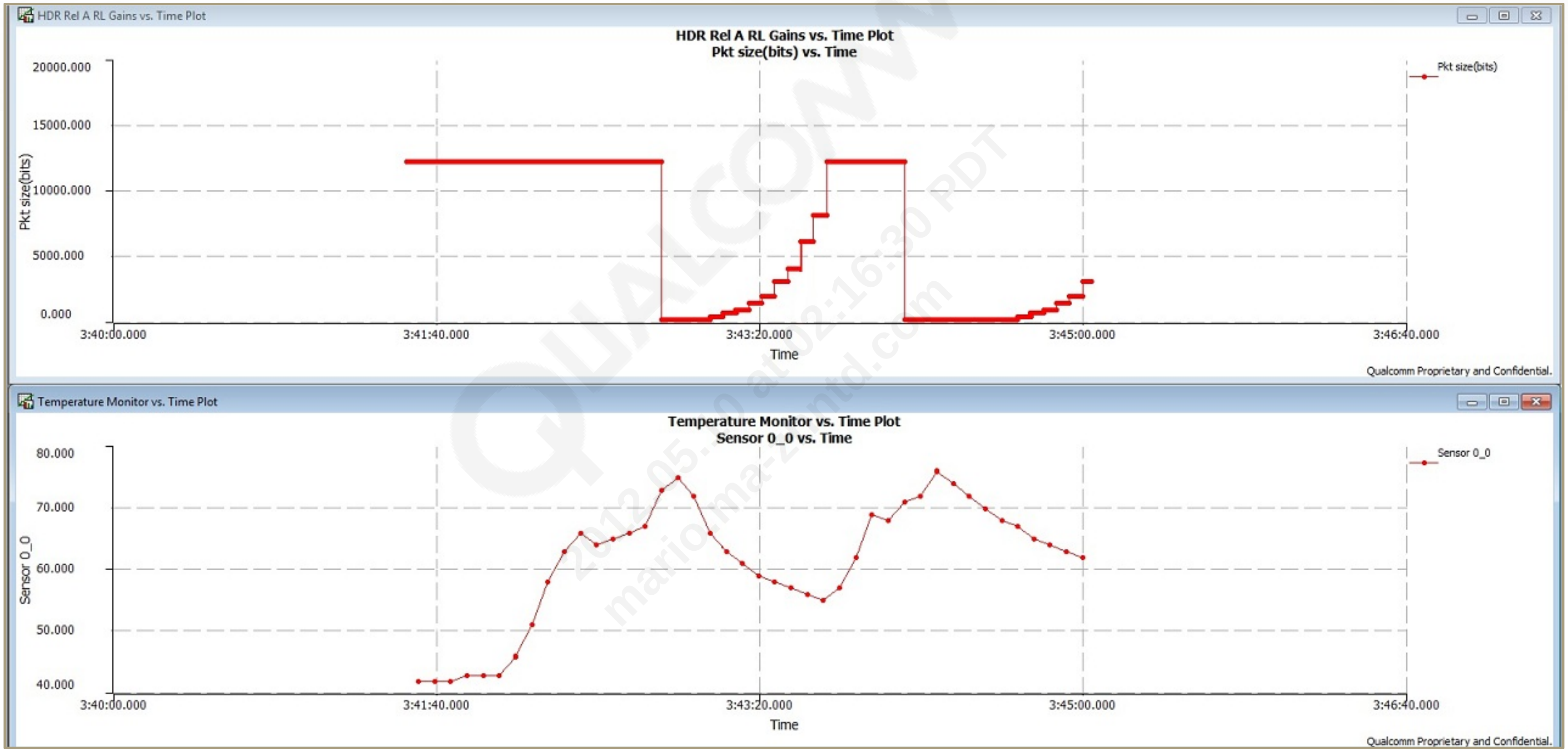
	Normal	Elevated	Severe	Critical
Modem/data throttle (only UL)	No restrictions	Start throttling upload data	Continue to throttle upload	Disable all data modes
Maximum Tx power throttling	Full power range	Full power range	Limit max Tx power	N/A (data modes already disabled)
Emergency/call shutdown	None	None	None	Call released/limited service; only E911 allowed



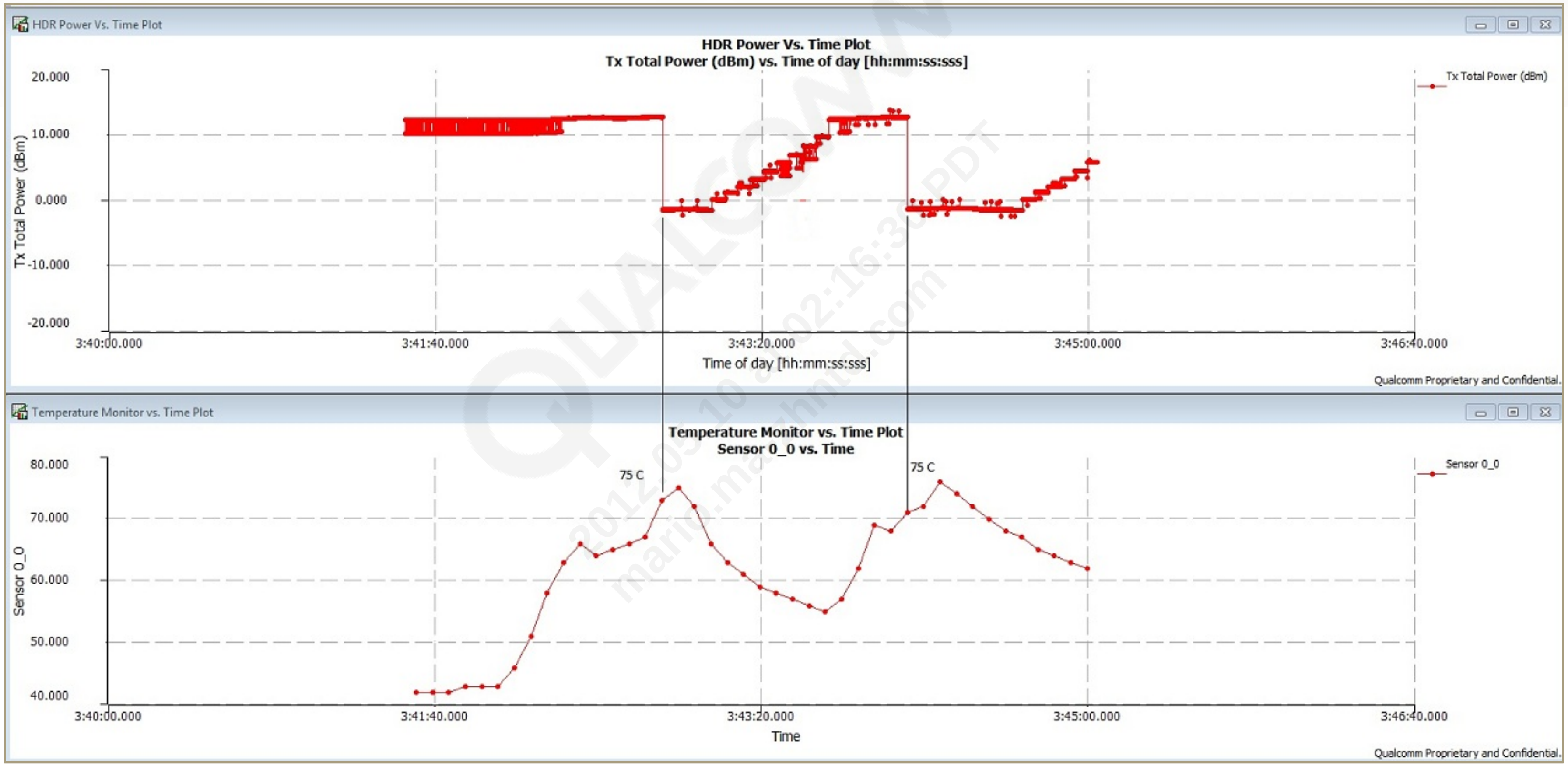
1X/EvDO Thermal Management Approach

- 1X thermal management approach
  - Once mitigation is initiated on 1X, the device will stop requesting for R-SCH (reverse supplemental channel), which will stop the reverse link traffic on 1X.
  - Upon coming to Normal Mitigation state, the device will resume R-SCH processing.
- EV-DO thermal management approach
  - Once mitigation is initiated:
    - 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.
    - In critical state, RMAC on 1xEV-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.

1xEV-DO Thermal Management Approach (cont.)



1xEV-DO Thermal Management Approach (cont.)



Debugging – Log Packets

- 1X log packets
  - 0x14D8 – Temperature monitor log
  - 0x1005 – Reverse Channel Traffic message
  - 0x1008 – Forward Channel Traffic message
  - Log packet [0x14D8] temperature monitor
  - MSG F3 [00043/02] flow controller
- EV-DO log packets
  - 0x14D8 – Temperature monitor log
  - 0x127D – 1xEV Rev-A RL gains
  - 0x1069 – 1xEV power
  - Log packet [0x14D8] temperature monitor
  - MSG F3 [00043/02] flow controller

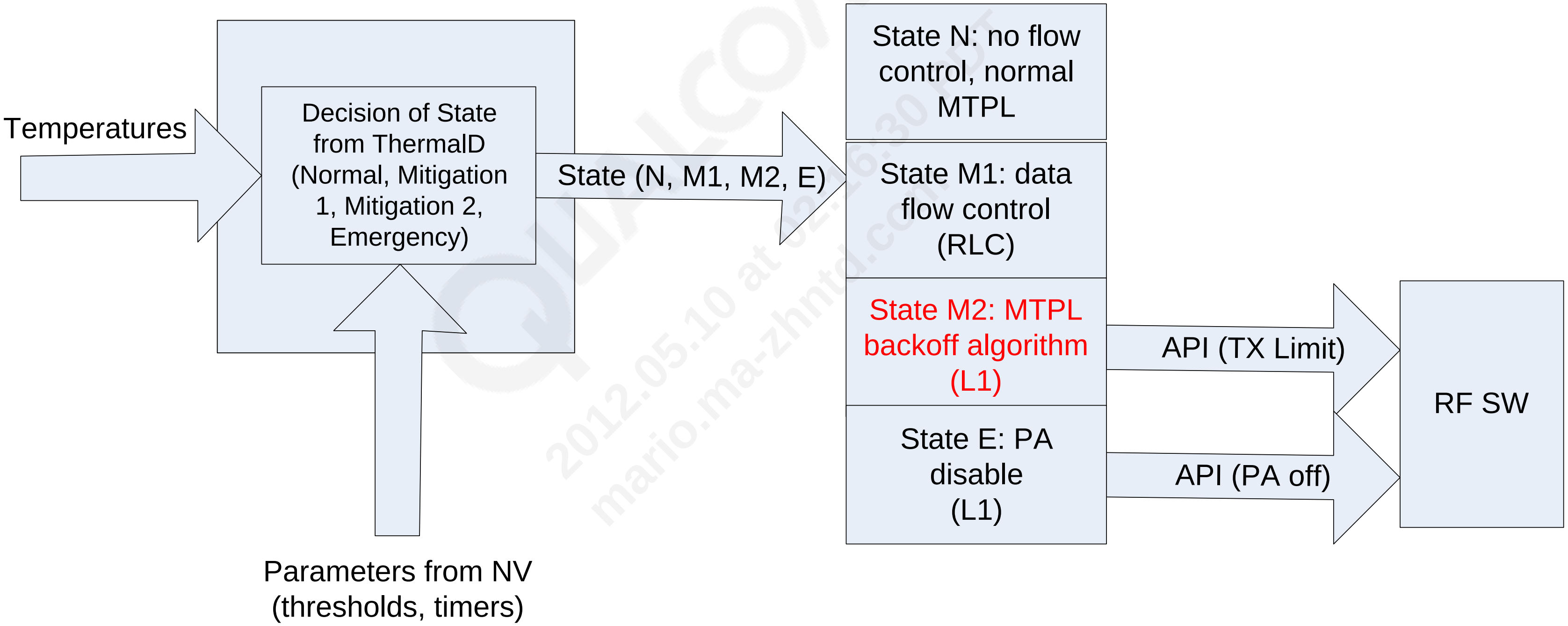
WCDMA TM

Thermal Management Actions Breakdown

- Following is an example, which can be configured differently.

	Normal	Elevated	Severe	Critical
Modem/data throttle (UL and DL)	No restrictions	Start throttling upload data	Throttle both upload and download	Disable all data modes
Maximum Tx power throttling	Full power range	Full power range	Limit maximum Tx power	N/A (data modes already disabled)
Emergency/call shutdown	None	None	None	Call released/limited service; only E911 allowed

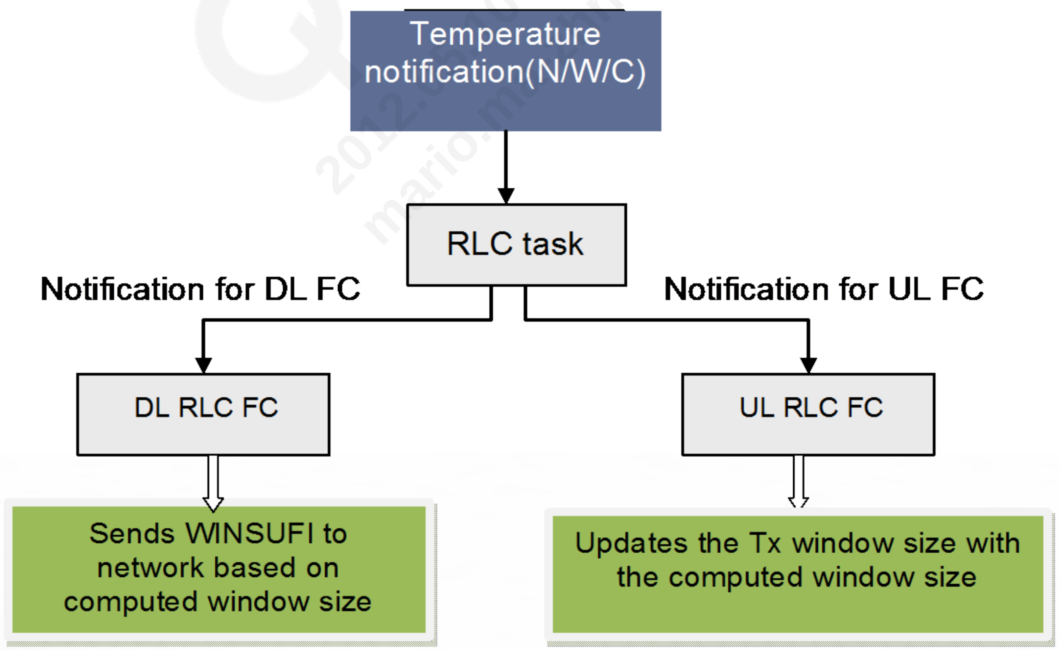
Thermal Management Flowchart



Mitigation Level 1 – Flow Control

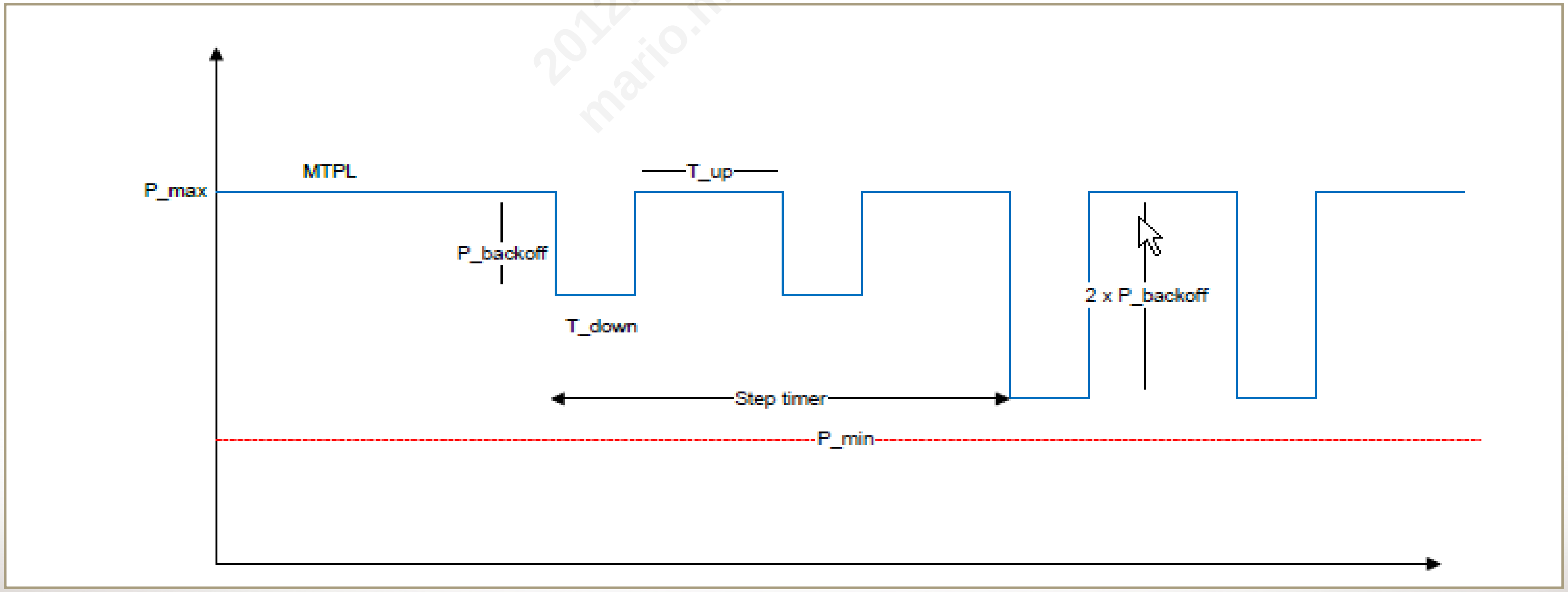
- Algorithm is only applied in:
  - UL direction by reducing the PDU size
  - DL direction by sending WIN_SIZE SUFI to network to reduce the DL flow
  - Relevant function with implementation

```
rlci_dl_fc_tx_new_win_sufi
 rlci_fc_new_win_size
```

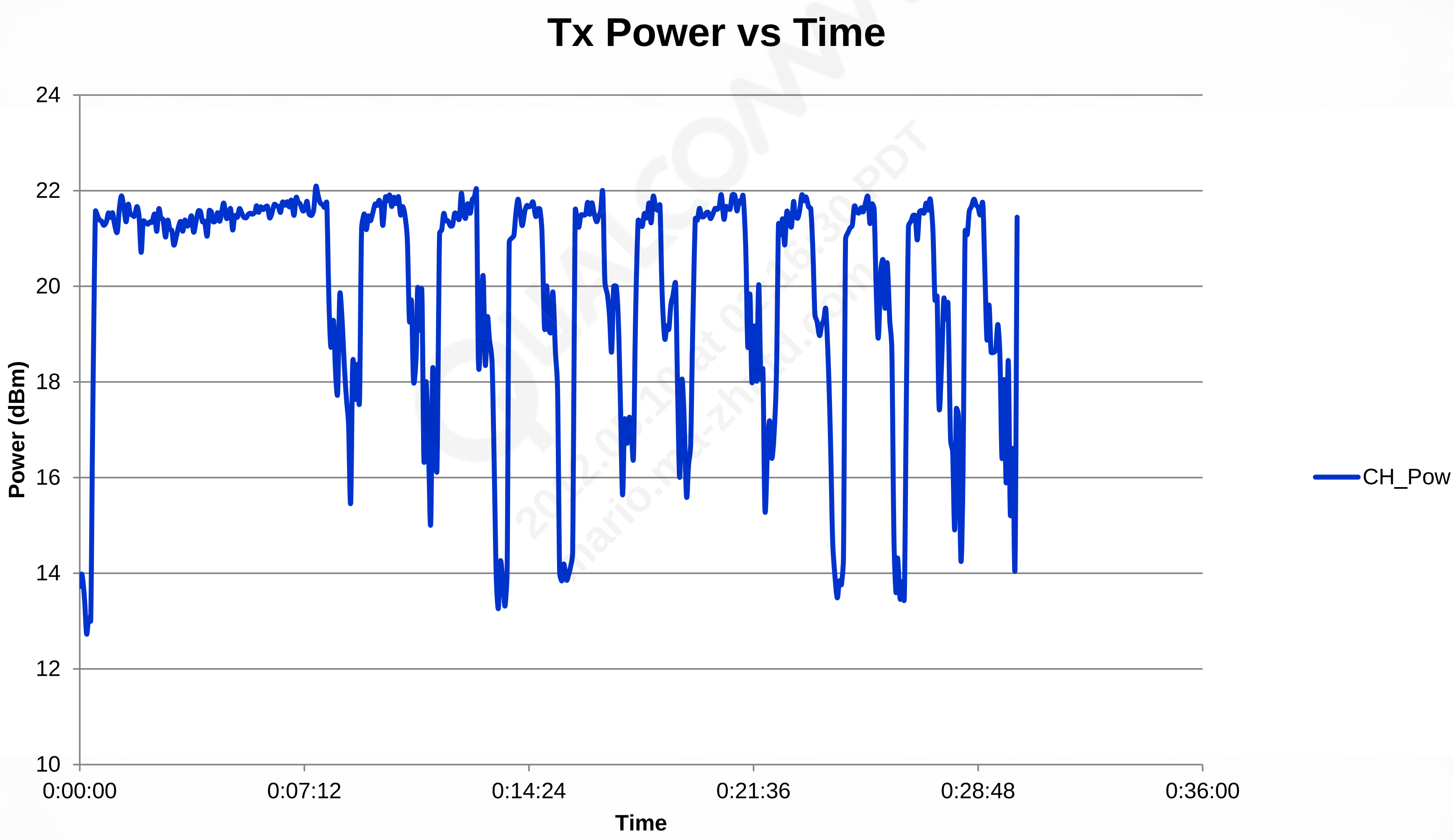


Mitigation Level 2 – Tx Power Backoff

- Algorithm is only applied if UE is in CELL_DCH
- Upon entering CELL_DCH, if UE is already in this Mitigation 2 state, L1 software shall apply the MTPL backoff algorithm immediately, based on the following configuration:
  - Initial MTPL Backoff Value – 5 dB
  - T_down timer for duty-cycle – 400 ms
  - T_up timer for duty-cycle – 50 ms
  - Step timer value – 10 sec



Mitigation Level 2 – Tx Power Backoff (cont.)



Mitigation Level 3 (Emergency)

- When this state is entered:
  - RRC connection is released
  - Devices camp in limited service
  - Allows only E911 call until the thermal mitigation level is reduced

Debugging: Log Packets

- The following log packets/F3 are required for debugging WCDMA thermal-related issues:
  - Log packet [0x14D8] temperature monitor
  - MSG F3 [00043/02] flow controller
  - RLC log packets

Debugging

- Relevant log excerpts

```
1980 Jan 6 00:29:08.971 [8D] 0x413B WCDMA RLC UL AM PDU
Number of AM UL Entities = 1
Entity[0]:
Data Logical Channel ID = 19
Number of PDUs Logged = 10
PDU Size (in Bits) = 336
PDU Log(s):
Raw: 0x01 0x7F 0xF2 0xC9
<-CONTROL PDU:: Chan:19, Type: STATUS
SUFI[0]: WINDOW SIZE => 2047 ← Sending WIN SIZE for max DL throughput


1980 Jan 6 00:29:14.794 [D4] 0x14D8 Temperature Monitor Log
Version = 1
Number Of Samples = 2
-----------------------------
| | |Sensor|Temperature|
|# |SSID|ID |Reading |
-----------------------------
| 0| 0| 0| 98| ← Temperature on PA is rising


1980 Jan 6 00:29:44.826 [8F] 0x14D8 Temperature Monitor Log
Version = 1
Number Of Samples = 2
-----------------------------
| | |Sensor|Temperature|
|# |SSID|ID |Reading |
-----------------------------
| 0| 0| 0| 99|  ← Temperature on PA is rising
```

Debugging (cont.)

```
1980 Jan 6 00:29:44.834 [90] 0x413B WCDMA RLC UL AM PDU
Number of AM UL Entities = 1
Entity[0]:
Data Logical Channel ID = 19
Number of PDUs Logged = 16
PDU Size (in Bits) = 336
PDU Log(s):
Raw: 0x01 0x36 0xB0 0x72
<-CONTROL PDU:: Chan:19, Type: STATUS
SUFI[0]: WINDOW SIZE => 875  ← Temperature on PA crossed the threshold and RLC FC kicked in. A new DL
WIN_SIZE is sent to reduce throughput

1980 Jan 6 00:29:54.835 [79] 0x14D8 Temperature Monitor Log
Version = 1
Number Of Samples = 2
-----------------------------
| | |Sensor|Temperature|
|# |SSID|ID |Reading |
-----------------------------
| 0| 0| 0| 99|  ← Temperature is still not below the clear threshold

1980 Jan 6 00:30:14.837 [49] 0x413B WCDMA RLC UL AM PDU
Number of AM UL Entities = 1
Entity[0]:
Data Logical Channel ID = 19
Number of PDUs Logged = 5
PDU Size (in Bits) = 336
PDU Log(s):
Raw: 0x01 0x11 0x10 0x72
<-CONTROL PDU:: Chan:19, Type: STATUS
SUFI[0]: WINDOW SIZE => 273  ← A new DL WIN_SIZE is sent to further reduce throughput
```

Debugging (cont.)

```
1980 Jan 6 00:30:34.857 [1C] 0x14D8 Temperature Monitor Log
Version = 1
Number Of Samples = 2
-----------------------------
| | |Sensor|Temperature|
|# |SSID|ID |Reading |
-----------------------------
| 0| 0| 0| 99|


1980 Jan 6 00:31:14.833 [BA] 0x413B WCDMA RLC UL AM PDU
Number of AM UL Entities = 1
Entity[0]:
Data Logical Channel ID = 19
Number of PDUs Logged = 8
PDU Size (in Bits) = 336
PDU Log(s):
Raw: 0x01 0x03 0x30 0x02
<-CONTROL PDU:: Chan:19, Type: STATUS
SUFI[0]: WINDOW SIZE => 51  ← A new DL WIN_SIZE is sent to further reduce throughput


1980 Jan 6 00:32:14.916 [34] 0x14D8 Temperature Monitor Log
Version = 1
Number Of Samples = 2
-----------------------------
| | |Sensor|Temperature|
|# |SSID|ID |Reading |
-----------------------------
| 0| 0| 0| 98| ← Temperature started reducing
```

Appendix – Thermal Characterization/Tuning

Thermal Tuning Requirement

- Key factors affecting thermal management and configuration
  - Different thermal characteristic of each form factor
  - Thermal hardware design
  - Targeted use case
  - Performance tradeoff
  - Ambience temperature

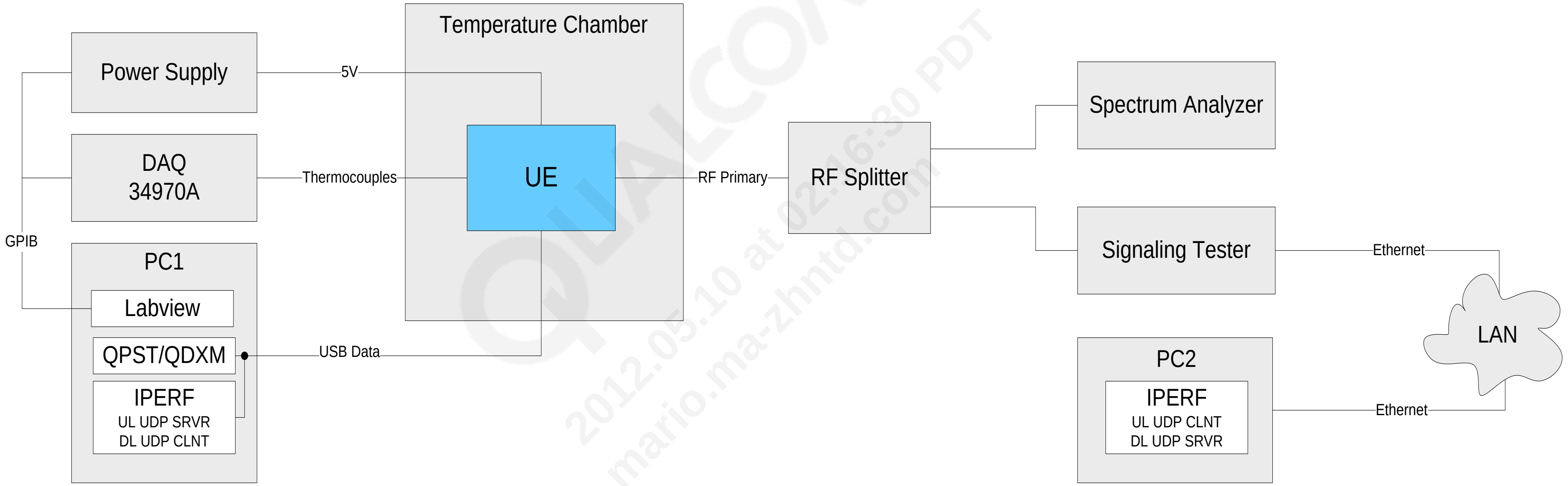
Thermal Tuning Setup Requirements

- Requirements
  - Infrared (IR) Camera
  - Thermocouples
  - Data acquisition unit for data logging from thermocouples
  - LTE call box
  - Needed applications (e.g., iperf, HD video, etc.) to run targeted use case
  - Fake battery to run long tests

Use Case Definition

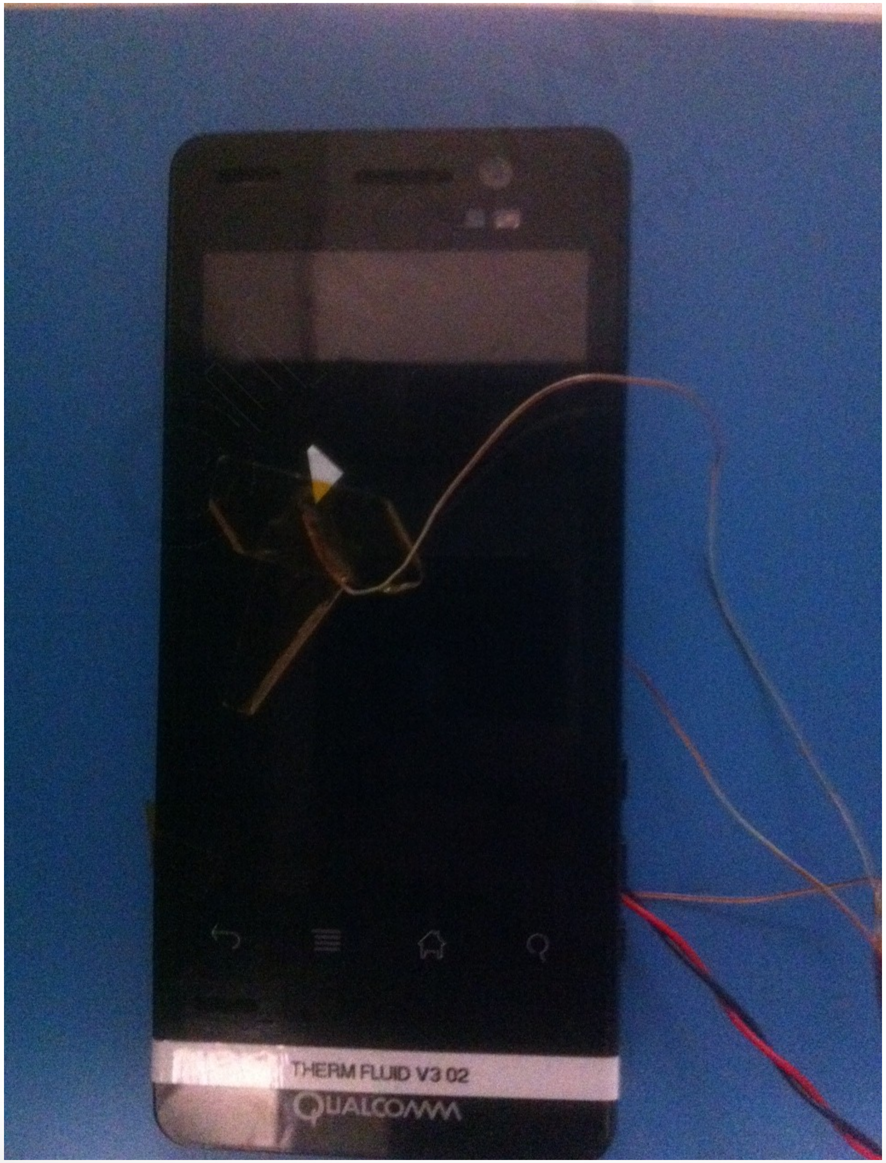
- Before initiating thermal tuning, a use case must be planned for which thermal characterization will be performed.
- Possible use cases are:
  - 3D gaming
  - 1080p encode
  - 1080p decode
  - LTE UL data call in poor RF
  - CPU intensive task
  - WLAN operation
  - Camera
  - Device charging
  - Combination of above use cases

Test Setup



Temperature Chamber
Power Supply
5V
DAQ
34970A
Thermocouples
UE
RF Primary
RF Splitter
Spectrum Analyzer
Signaling Tester
Ethernet
GPIB
PC1
Labview
QPST/QDXM
USB Data
IPERF
UL UDP SRVR
DL UDP CLNT
LAN
PC2
IPERF
UL UDP CLNT
DL UDP SRVR
Ethernet

Test Setup (Front HotSpot Thermocouple)



Test Setup (Back/POP/PA HotSpot Thermocouple)

Steps to Tune a Device

1. Run the test scenario of your choice with initial thermald.conf (i.e., thermald.conf with no actions defined), (check initial thermald.conf in the following slides).
2. After 20 min, monitor front of the device with thermal camera and find the hotspot.
3. Flip device and find hotspot on the back of the device.
4. Cool the device to room temperature between each test.
5. Run LTE 20 Mbps @ 20 dBm UL data call using iperf and find device hotspot.
6. Disassemble device and mount one thermocouple on center of pop memory package.
7. Reassemble device and place thermocouple on front hotspot and back hotspot.
8. Place the fourth thermocouple on the LTE UL data test hotspot.
9. For thermal analysis, the following tests should be run for 40 to 60 min or until crash when done without management.

Steps to Tune a Device (cont.)

10. Log thermocouple temperatures and thermal daemon data until the pop memory temperature passes 85°C or the front and back of the phone temperature rises more than acceptable limits.
11. Run test scenario + LTE 20 Mbps @ 20 dBm test case with initial configuration file. Log thermocouple temperatures and thermal daemon data until the pop memory temperature passes 85°C. Logging of TSENS sensors and PA sensors can be done using logcat logs.
12. Compare tsens temperature data against pop memory thermocouple. Find tsens sensor that matches closest to pop memory (typically tsens3 or tsens 4, however ID may affect which sensor tracks best) for the above two test cases.

Steps to Tune a Device (cont.)

13. Use following thresholds and actions on the sensor discovered in above step, which follows the pop memory. Use default thermald.conf (see appendix) file to make the changes.

```
sampling        1000
thresholds      75     78     81     84     87     90
thresholds_clr  72     75     78     81     84     87
actions         cpu     cpu     cpu     cpu     cpu     shutdown
action_info     1296000    1188000    918000    756000    648000    5000
```

14. Rerun tests with the configuration file created above. Log thermocouple temperatures and thermal daemon data until temperature stabilizes or the pop memory temperature passes 85°C or case temperature reaches above acceptable limits.
15. If pop memory temp crosses 85°C or case temperature rises above limits in test time, reduce thresholds to limit performance at lower temperatures than as specified in default_config.conf. Both the threshold temperature, as well as CPU frequency can be varied as needed. Repeat step 11 and further optimize the thermal parameters in config file.
16. Repeat steps 11 and 12 for other use cases to validate/fine tune configurations.

Steps to Tune Modem Mitigation Thresholds

1. Modem mitigation should be configured on PA temperature sensors to control the heat generated in modem related scenarios.
2. Three levels of mitigation actions are available for modem mitigation:
   - Level 1 – Uplink data throttling
   - Level 2 – Tx power backoff
   - Level 3 – Emergency call
3. We recommend to use modem mitigation as the last level of mitigation when CPU mitigation is not enough to meet temperature specifications under extreme concurrency scenarios.
4. For tuning purpose, select a modem concurrency scenario which involves LTE data call (uplink + downlink) along with CPU/GPU intensive application.
5. Start with the following configuration on the available PA sensors:

```
[pa_therm0]
sampling         1000
thresholds       70     80      90
thresholds_clr   65     75      85
actions          modem     modem     modem
action_info      1     2     3
```

Steps to Tune Modem Mitigation Thresholds (cont.)

6. Set up the phone with thermald.conf file with the PA thresholds from 5 and the CPU mitigation thresholds and actions derived from CPU tuning procedure described previously.
7. Additionally, set up the thermocouple measurement as done previously for CPU tuning.
8. Establish a LTE data call with continuous data streaming happening in both uplink and downlink direction.
9. Start a CPU/GPU related application. You can reuse the application used for CPU thermal tuning.
10. Let the phone continue in this scenario until CPU or modem mitigation begins while observing the trend of touch and tsens temperature.
11. If the CPU mitigation by itself is able to maintain the touch temperature below the specification limit, then no additional tuning is required for modem mitigation.
12. If touch temperature limits are exceeded, then lower the thresholds on PA by 3°C and repeat the experiment until the spec is met.

Initial thermald.conf for Finding HotSpots

```
sampling         1000
[pa_therm0]
sampling         1000
thresholds       65
thresholds_clr   60
actions          none
action_info      0

[pa_therm1]
sampling         1000
thresholds       65
thresholds_clr   60
actions          none
action_info      0

[tsens_tz_sensor0]
sampling         1000
thresholds       30
thresholds_clr   27
actions          none
action_info      0

[tsens_tz_sensor1]
sampling         1000
thresholds       30
thresholds_clr   27
actions          none
action_info      0
```

Initial thermald.conf for Finding HotSpots (cont.)

```
[tsens_tz_sensor2]
sampling          1000
thresholds        30
thresholds_clr    27
actions           none
action_info       0

[tsens_tz_sensor3]
sampling          1000
thresholds        30
thresholds_clr    27
actions           none
action_info       0

[tsens_tz_sensor4]
sampling          1000
thresholds        30
thresholds_clr    27
actions           none
action_info       0
```

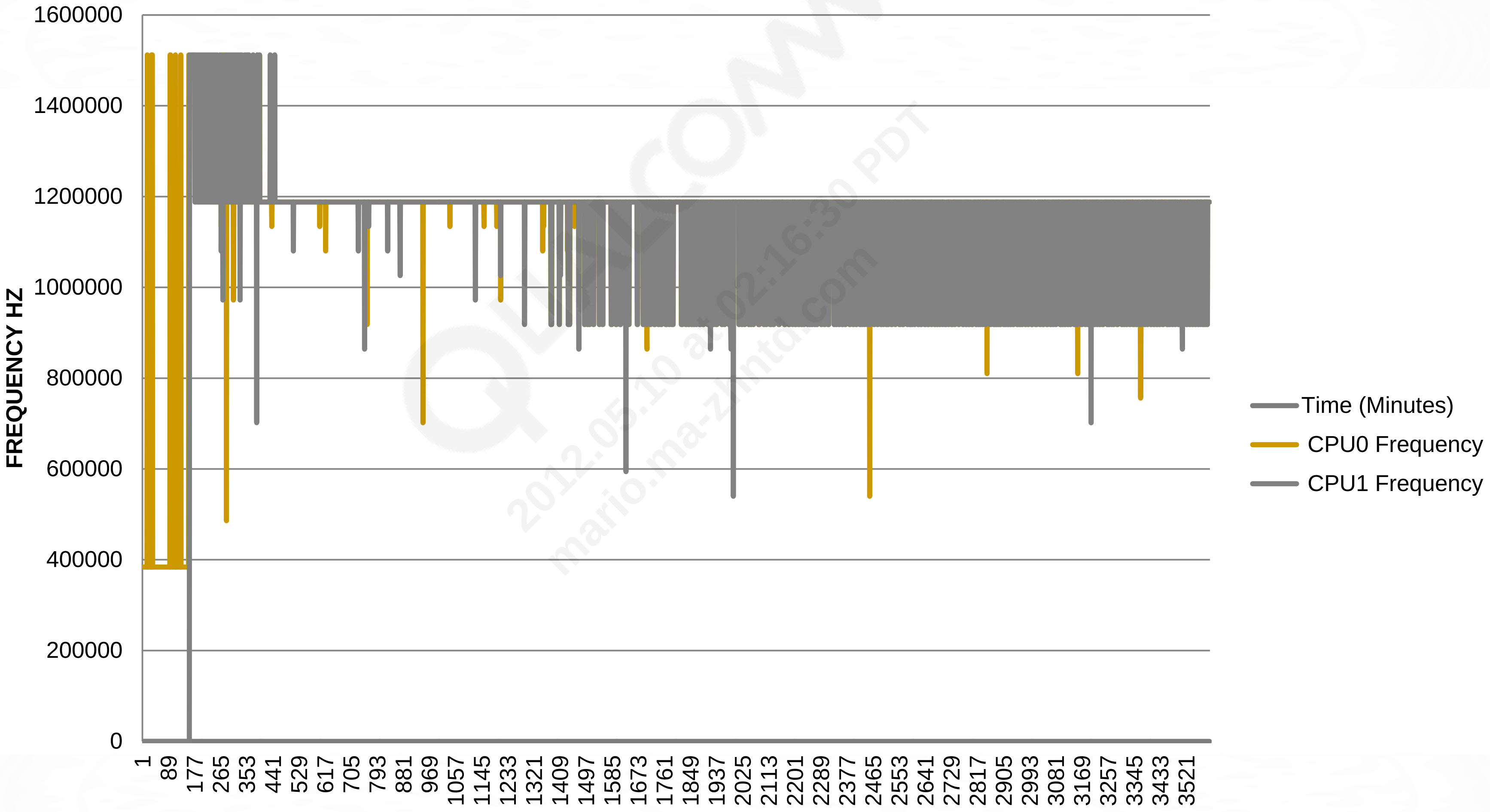
Default thermald.conf File for Tuning

```
debug
sampling          5000

[pa_therm0]
sampling          1000
thresholds        70    80     90
thresholds_clr    65    75     85
actions           modem    modem    modem
action_info       1    2    3

[tsens_tz_sensor0]
sampling          1000
thresholds        65      90      93      96      99      102     105
thresholds_clr    62      87      90      93      96      99      102
actions           cpu     cpu     cpu     cpu     cpu     cpu     shutdown
action_info       1512000 1296000 1188000 918000  756000  648000  5000

[tsens_tz_sensor1]
sampling          1000
thresholds        75
thresholds_clr    72
actions           none
action_info       0
```

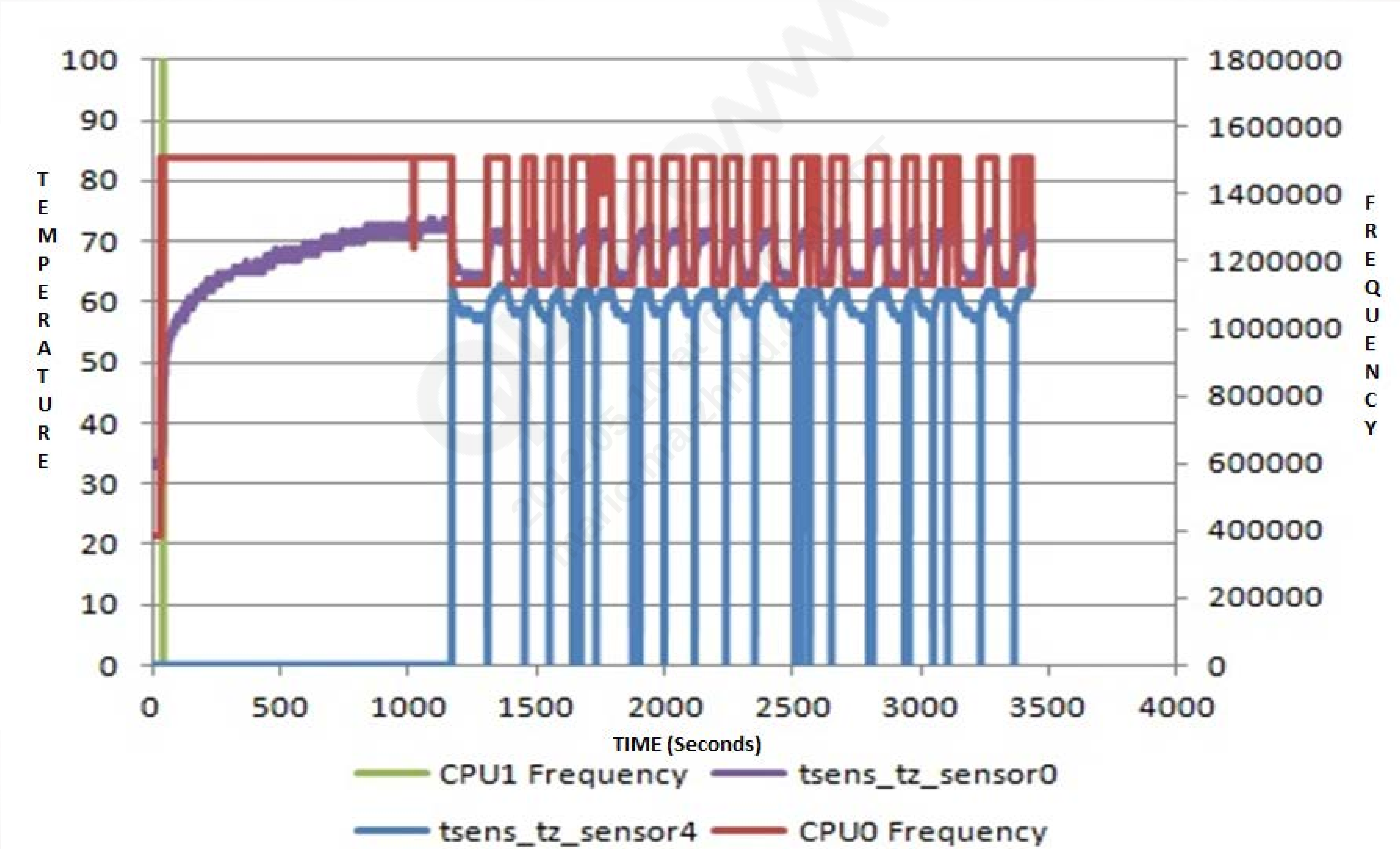
Default thermald.conf File for Tuning (cont.)

```
[tsens_tz_sensor2]
sampling          1000
thresholds        75
thresholds_clr    72
actions           none
action_info       0

[tsens_tz_sensor3]
sampling          1000
thresholds        75
thresholds_clr    72
actions           none
action_info       0


[tsens_tz_sensor4]
sampling          1000
thresholds        75
thresholds_clr    72
actions           none
action_info       0
```


Example – Before Tuning (Conservative/Default Config)



Example – After Tuning (With Final/Tuned Config)



Sample FFA/MTP thermald.conf File (After Tuning)

```
debug
sampling         5000

[pa_therm0]
sampling         1000
thresholds       70    80     90
thresholds_clr   65    75     85
actions          modem    modem    modem
action_info      1    2    3

[tsens_tz_sensor0]
sampling         1000
thresholds       65      90      93      96      99      102     105
thresholds_clr   62      87      90      93      96      99      102
actions          cpu     cpu     cpu     cpu     cpu     cpu     shutdown
action_info      1512000 1296000 1188000 918000  756000  648000  5000

[tsens_tz_sensor1]
sampling         1000
thresholds       75
thresholds_clr   72
actions          none
action_info      0
```


Sample FFA/MTP thermald.conf File (After Tuning) (cont.)

```
[tsens_tz_sensor2]
sampling          1000
thresholds        75
thresholds_clr    72
actions           none
action_info       0

[tsens_tz_sensor3]
sampling          1000
thresholds        75      78      81      84      87      90
thresholds_clr    72      75      78      81      84      87
actions           cpu     cpu     cpu     cpu     cpu     shutdown
action_info       1296000 1188000 918000  756000  648000  5000

[tsens_tz_sensor4]
sampling          1000
thresholds        75
thresholds_clr    72
actions           none
action_info       0
```

References

Ref.	Document	
Qualcomm		
Q1	*Application Note: Software Glossary for Customers*	CL93-V3077-1
Q2	*Thermal Design Checklist*	80-VU794-21
Q3	*Thermal Management of MSM8660™/MSM8260™/APQ8060 Devices*	80-VU872-16
Q4	*Thermal Protection Algorithm Overview*	80-VT344-1
Q5	*MDM8200™ Thermal Protection Algorithm Application Note*	80-VJ372-14
Q6	*Thermal Protection Algorithm Overview*	80-VT344-1
Q7	*Application Note: MDM9600™ Thermal Protection Algorithm Details*	80-VP146-15
Q8	*Application Note: MDM9200™ Thermal Protection Algorithm Details*	80-VP145-15
Q9	*Application Note: MDM8220™ Thermal Protection Algorithm Details*	80-VP144-15
Q10	*MSM8960™ Thermal Mitigation Algorithm*	80-N8633-1
Q11	*MDM9x15 Thermal Mitigation Algorithm*	80-N8633-2
Q12	*Introduction to Mobile Devices Thermal Design – App Note*	80-VU794-11

Questions?

https://support.cdmatech.com