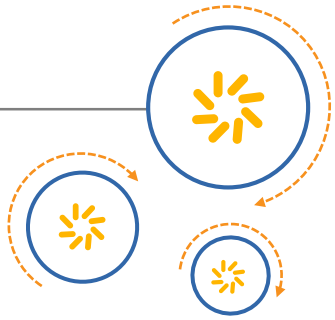
Qualcomm Technologies, Inc.



Modem Clock and Power Manager (MCPM)

Debug Guide

80-NR497-1 C

August 20, 2015

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

Revision history

Revision	Date	Description
A	August 2014	Initial release
B	June 2015	Added Section 3.3
C	August 2015	Added Section 3.4

Contents

Figures

1 Introduction

1.1 Purpose

This document provides software engineers with the necessary information for debugging the Modem Clock and Power Manager (MCPM).

1.2 Conventions

Function declarations, function names, type declarations, attributes, and code samples appear in a different font, for example, `#include`.

Code variables appear in angle brackets, for example, `<number>`.

Commands to be entered appear in a different font, for example, **`copy a:*.* b:`**.

Button and key names appear in bold font, for example, click **Save** or press **Enter**.

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

1.3 Technical assistance

For assistance or clarification on information in this document, submit a case to Qualcomm Technologies, Inc. (QTI) at https://createpoint.qti.qualcomm.com/.

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

2 MCPM overview

MCPM is the software module used by all modem technology L1s for enabling clocks. It resides on the modem software processor and performs state-based (e.g., Inactive, Idle, Voice, etc.) clock and power configuration. For each modem state, MCPM configures both modem-specific and MPSS external resources.

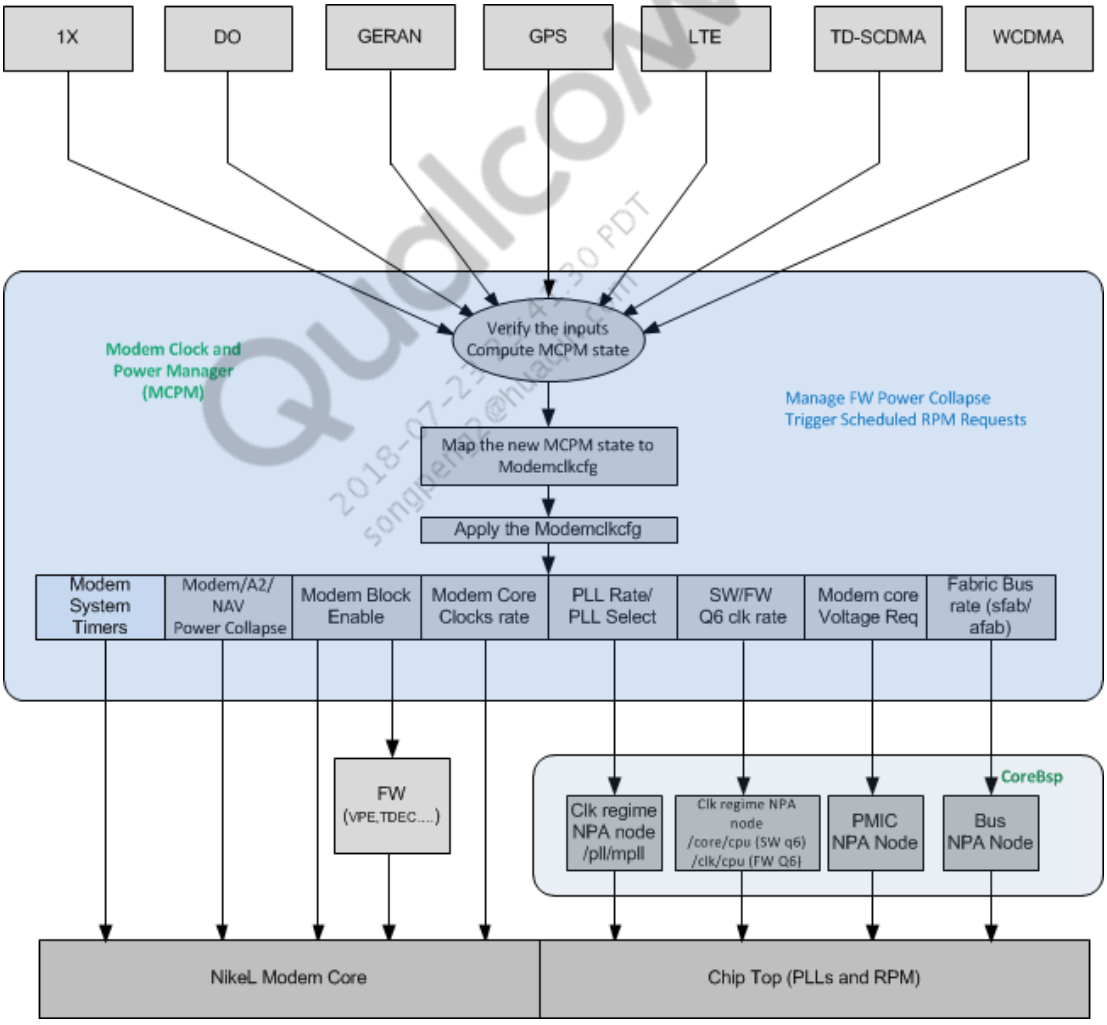


Figure 2-1 MCPM software module example

MCPM manages the following in a power efficient manner:

- Modem PLLs
- Modem clock enable and disable (TDEC, VPE, etc.)
- Modem power gating (power collapse)
- Modem core clock rates (offline, Tx, GSM, etc.)
- Modem core voltage switching (SVS/NOM/turbo) requests
- Enabling of RTC/STMRs for neighbor measurements
- Hexagon™ processor clock speed
- Requests for system resources:
  - MSM8960 – SFAB, AFAB, DDR
  - MSM8974, MSM8994 – SNOC, BIMC
- Hexagon firmware power collapse (MSM8960-only)
- Hexagon software power collapse
- Clock for DATA calls:
  - MSM8960 – Daytona FAB
  - MSM8974, MSM8994 – PNOC

3 Check MCPM settings in the F3 log

MCPM is released as a library; therefore, MCPM debugging is limited on the customer side. The F3 log is the only way to check the MCPM setting.

3.1 MSM8974

3.1.1 Check MCPM setting

Search the F3 log for MCPM CONFIG MODEM:

```
03:27:20.582     2366   mcpm.c   MCPM CONFIG MODEM start for Req: 37, time
= 1d930f65  // MCPM_LTE_START_REQ, XOTimeTick
03:27:20.585     2373   mcpm.c   MCPM CONFIG MODEM end for Req: 37, time =
1d93de4e, duration = 2758 uSec // MCPM_LTE_START_REQ, XOTimeTick, Duration
```

If there is no MCPM message in the F3 log, there has been no recent MCPM operation.

The request type is defined in \modem_proc\mpower\api\mcpm_api.h and can differ from Product Line (PL) to PL. The following example is from MPSS.DI.2.0 (Dime 2.0):

```
typedef enum
{
  /* 1X Requests */
  MCPM_1X_START_REQ                                = 0,
  MCPM_1X_STOP_REQ                                 = 1,
  MCPM_1X_WAKE_UP_REQ                              = 2,
  MCPM_1X_GO_TO_SLEEP_REQ                          = 3,
  MCPM_1X_GO_TO_PSEUDO_SLEEP_REQ                   = 4,
  MCPM_1X_IDLE_REQ                                 = 5,
  MCPM_1X_VOICE_REQ                                = 6,
  MCPM_1X_DATA_REQ                                 = 7,
  MCPM_1X_PARMS_UPDATE_REQ                         = 8,

  /* DO Requests */
  MCPM_DO_START_REQ                                = 9,
  MCPM_DO_STOP_REQ                                 = 10,
  MCPM_DO_WAKE_UP_REQ                              = 11,
  MCPM_DO_GO_TO_SLEEP_REQ                          = 12,
```

```
MCPM_DO_IDLE_REQ                                = 13,
MCPM_DO_START_DATA_REQ                          = 14,
MCPM_DO_STOP_DATA_REQ                           = 15,
MCPM_DO_PARMS_UPDATE_REQ                        = 16,

/* GERAN Requests */
MCPM_GERAN_START_REQ                            = 17,
MCPM_GERAN_STOP_REQ                             = 18,
MCPM_GERAN_WAKE_UP_REQ                          = 19,
MCPM_GERAN_GO_TO_SLEEP_REQ                      = 20,
MCPM_GERAN_IDLE_REQ                             = 21,
MCPM_GERAN_VOICE_START_REQ                      = 22,
MCPM_GERAN_VOICE_STOP_REQ                       = 23,
MCPM_GERAN_DATA_START_REQ                       = 24,
MCPM_GERAN_DATA_STOP_REQ                        = 25,
MCPM_GERAN_PARMS_UPDATE_REQ                     = 26,

/* GERAN1 Requests */
MCPM_GERAN1_START_REQ                           = 27,
MCPM_GERAN1_STOP_REQ                            = 28,
MCPM_GERAN1_WAKE_UP_REQ                         = 29,
MCPM_GERAN1_GO_TO_SLEEP_REQ                     = 30,
MCPM_GERAN1_IDLE_REQ                            = 31,
MCPM_GERAN1_VOICE_START_REQ                     = 32,
MCPM_GERAN1_VOICE_STOP_REQ                      = 33,
MCPM_GERAN1_DATA_START_REQ                      = 34,
MCPM_GERAN1_DATA_STOP_REQ                       = 35,
MCPM_GERAN1_PARMS_UPDATE_REQ                    = 36,

/* LTE Requests */
MCPM_LTE_START_REQ                              = 37,
MCPM_LTE_STOP_REQ                               = 38,
MCPM_LTE_ACQ_REQ                                = 39,
MCPM_LTE_WAKE_UP_REQ                            = 40,
MCPM_LTE_GO_TO_SLEEP_REQ                        = 41,
MCPM_LTE_IDLE_REQ                               = 42,
MCPM_LTE_DATA_START_REQ                         = 43,
MCPM_LTE_TDD_DATA_START_REQ                     = 44,
MCPM_LTE_FDD_DATA_START_REQ                     = 45,
MCPM_LTE_TDD_VOLTE_DATA_START_REQ               = 46,
MCPM_LTE_FDD_VOLTE_DATA_START_REQ               = 47,
MCPM_LTE_GO_TO_LIGHT_SLEEP_REQ                  = 48,
MCPM_LTE_GO_TO_LONG_LIGHT_SLEEP_REQ             = 49,
MCPM_LTE_GO_TO_LIGHT_SLEEP_NO_MODEM_FREEZE_REQ  = 50,
```

```
  MCPM_LTE_DATA_STOP_REQ                          = 51,
  MCPM_LTE_PARMS_UPDATE_REQ                       = 52,

  /* TDSCDMA Requests */
  MCPM_TDSCDMA_START_REQ                          = 53,
  MCPM_TDSCDMA_STOP_REQ                           = 54,
  MCPM_TDSCDMA_ACQ_REQ                            = 55,
  MCPM_TDSCDMA_WAKE_UP_REQ                        = 56,
  MCPM_TDSCDMA_GO_TO_SLEEP_REQ                    = 57,
  MCPM_TDSCDMA_IDLE_REQ                           = 58,
  MCPM_TDSCDMA_VOICE_START_REQ                    = 59,
  MCPM_TDSCDMA_VOICE_STOP_REQ                     = 60,
  MCPM_TDSCDMA_DATA_START_REQ                     = 61,
  MCPM_TDSCDMA_DATA_STOP_REQ                      = 62,
  MCPM_TDSCDMA_PARMS_UPDATE_REQ                   = 63,

  /* WCDMA Requests */
  MCPM_WCDMA_START_REQ                            = 64,
  MCPM_WCDMA_STOP_REQ                             = 65,
  MCPM_WCDMA_WAKE_UP_REQ                          = 66,
  MCPM_WCDMA_GO_TO_SLEEP_REQ                      = 67,
  MCPM_WCDMA_IDLE_REQ                             = 68,
  MCPM_WCDMA_VOICE_START_REQ                      = 69,
  MCPM_WCDMA_VOICE_STOP_REQ                       = 70,
  MCPM_WCDMA_DATA_START_REQ                       = 71,
  MCPM_WCDMA_DATA_STOP_REQ                        = 72,
  MCPM_WCDMA_PARMS_UPDATE_REQ                     = 73,

  /* GPS Requests */
  MCPM_GPS_STOP_REQ                               = 74,
  MCPM_GPS_ACQ_REQ                                = 75,
  MCPM_GPS_NON_DPO_REQ                            = 76,
  MCPM_GPS_DPO_ON_REQ                             = 77,
  MCPM_GPS_PARMS_UPDATE_REQ                       = 78,

/* RF Requests */
  MCPM_RF_START_REQ                               = 79,
  MCPM_RF_STOP_REQ                                = 80,

  /* A2 Requests */
  MCPM_A2_START_REQ                               = 81,
  MCPM_A2_STOP_REQ                                = 82,

  /* GSM CIPHERING Requests */
```

```
  MCPM_GSM_CIPHERING_START_REQ                    = 83,
  MCPM_GSM_CIPHERING_STOP_REQ                     = 84,

    /* GSM1 CIPHERING Requests */
  MCPM_GSM1_CIPHERING_START_REQ                   = 85,
  MCPM_GSM1_CIPHERING_STOP_REQ                    = 86,

  MCPM_TECH_MAX_REQ
} mcpm_request_type;
```

3.1.2 Hexagon clock

Search for MCPMDRV Request Hexagon software:

```
03:27:20.583     1004   mcpm_drv_npa.c   MCPM_DRV_NPA: MCPMDRV Request Q6SW
384000   // Q6 clock request, 384MHz
03:27:20.585      266   mcpm_npa_resrc.c   MCPM_NPA_RESRC: Updating CLKCPU
with state 384000
03:27:20.585     1374      mcpm_drv_mux.c:mcpm local current running state
of resources: clk_bus = 72000 KHz, VPE = 288000 KHz, Q6 = 384000 KHz MCVS
params here VPE = 0 Q6 = 0 clk_bus = 0 MP_HW_read :0x4 // mss_bus_cfg, VPE,
Q6 clock
```

The last requested clock is the active Hexagon clock. It is found in the NPA dump:

```
npa_resource (name: "/mcpm_q6clk/clk/cpu") (handle: 0xAF9DB68) (sequence:
0x00037C00) (units: MHz) (resource max: 576000) (active max: 576000)
(active state: 384000)  (active headroom: -192000) (request state: 384000)
(resource attributes: 0x00000000) (node_lock: 0xAF6D820) (event_lock:
0xAEBF430) (transaction: 0x0) ((_internal: 0xAF52A38) (dependent_state:
0x3) (driver_dur: ((min: 131) (min_time: 0x155EB8F9) (max: 4070) (max_time:
0x2DAAA4FD7) (total time: 980297) (count: 884) (avg: 1108)))))
```

3.1.3 CX, MX, MSS

- CX – MSM core voltage
- MX – MEM voltage for the internal memory
- MSS – Hexagon core voltage

Search for MCPM Voltage Level. The aggregated CX, MX, and MSS are printed.

```
03:27:20.585     1379   mcpm_drv_mux.c   MCPM Voltage Level for
Q6_VPE_CLK_bus req MSS 2 MX 0 CX 3, tech 5  // requested NPA for
MCPM_MSS_SVS, MCPM_MX_SVS, MCPM_CX_SVS, LTE
```

```
03:27:20.585      785   mcpm_drv_mux.c   MCPM Voltage Level MSS 2 MX 0 CX
3, tech 5
```

Type definition of CX:

```
typedef enum
{
  MCPM_CX_NO_VOTE = 0,
  MCPM_CX_VOL_RETENTION = 1,
  MCPM_CX_VOL_LOW_MINUS = 2,
  MCPM_CX_SVS = 3,
  MCPM_CX_NOM = 4,
  MCPM_CX_VOL_NOM_PLUS = 5,
  MCPM_CX_TURBO = 6,
  MCPM_CX_MAX
} MCPM_CX_vol_level;
```

Type definition of MX:

```
typedef enum
{
  MCPM_CX_NO_VOTE = 0,
  MCPM_CX_VOL_RETENTION = 1,
  MCPM_CX_VOL_LOW_MINUS = 2,
  MCPM_CX_SVS = 3,
  MCPM_CX_NOM = 4,
  MCPM_CX_VOL_NOM_PLUS = 5,
  MCPM_CX_TURBO = 6,
  MCPM_CX_MAX
} MCPM_CX_vol_level;
```

Type definition of MSS:

```
typedef enum
{
  MCPM_MSS_NO_VOTE = 0,

/* Shouldn't be used by techs. This corner is for eLDO */
  MCPM_MSS_SVS2 = 1,
  MCPM_MSS_SVS = 2,

  /* Only used by modem for vpe@384MHz */
  MCPM_MSS_SVS_PLUS = 3,
```

```
  MCPM_MSS_NOM = 4,
  MCPM_MSS_TURBO = 5,
  MCPM_MSS_SUPER_TURBO = 6,
  MCPM_MSS_MAX
} MCPM_MSS_vol_level;
```

The TECH type is defined in \modem_proc\mpower\api\mcpm_api.h and can differ from PL to PL. The following example is from Dime 2.0:

```
typedef enum
{

  /* MCPM 1X technolgy definition */
  MCPM_1X_TECH,

  /* MCPM GERAN technolgy definition */
  MCPM_GERAN_TECH,

 /* MCPM GERAN1 technolgy definition */
  MCPM_GERAN1_TECH,

  /* MCPM DO technolgy definition */
  MCPM_DO_TECH,

  /* MCPM WCDMA technolgy definition */
  MCPM_WCDMA_TECH,

  /* MCPM LTE technolgy definition */
  MCPM_LTE_TECH,

  /* MCPM TDSCDMA technolgy definition */
  MCPM_TDSCDMA_TECH,

  /* MCPM GPS technolgy definition */
  MCPM_GPS_TECH,

  /* MCPM RF technology definition */
  MCPM_RF_TECH,

  /* MCPM GSM CIPHERING definition */
  MCPM_GSM_CIPHERING_TECH,

  /* MCPM GSM1 CIPHERING definition */
  MCPM_GSM1_CIPHERING_TECH,
```

```
  /* MCPM A2 technolgy definition */
  MCPM_A2_TECH,

  /* number fo techs used for boundary checks  */
  MCPM_NUM_TECH
}
mcpm_tech_type;
```

3.1.4 Hexagon firmware power collapse and wake-up

There is no firmware power collapse on MSM8974 since the software and firmware are running on the same Hexagon. Search for MCPM FW SLEEP:

```
03:27:20.585      457   mcpm_saw.c MCPM FW SLEEP Finalized
```

Search for MCPM FW WAKE-UP:

```
03:27:20.585      560   mcpm_saw.c   MCPM FW WAKE-UP Complete
```

3.2 MDM9x35/MSM8994

3.2.1 Check MCPM setting

Search for MCPM_Config_Modem:

```
22:31:25.956:                              mcpm.c:2377     MCPM_Config_Modem:
Start of request: 81, begin time = 0x23b9966661.  // MCPM_A2_START_REQ

22:31:25.956:                              mcpm.c:2403     MCPM_Config_Modem:
End of request: 81, end time = 0x23b9969681, duration = 641 uSec.  // //
MCPM_A2_START_REQ
```

- Request – mcpm_request_type
- Time – XOTimeTick
- Duration – Configuration duration

The request type is defined in \modem_proc\mpower\api\mcpm_api.h and can differ from PL to PL.

```
typedef enum
{
  /* 1X Requests */
  MCPM_1X_START_REQ                                = 0,
  MCPM_1X_STOP_REQ                                 = 1,
  MCPM_1X_WAKE_UP_REQ                              = 2,
  MCPM_1X_GO_TO_SLEEP_REQ                          = 3,
  MCPM_1X_GO_TO_PSEUDO_SLEEP_REQ                   = 4,
  MCPM_1X_IDLE_REQ                                 = 5,
  MCPM_1X_VOICE_REQ                                = 6,
  MCPM_1X_DATA_REQ                                 = 7,
  MCPM_1X_PARMS_UPDATE_REQ                         = 8,

  /* DO Requests */
  MCPM_DO_START_REQ                                = 9,
  MCPM_DO_STOP_REQ                                 = 10,
  MCPM_DO_WAKE_UP_REQ                              = 11,
  MCPM_DO_GO_TO_SLEEP_REQ                          = 12,
  MCPM_DO_IDLE_REQ                                 = 13,
  MCPM_DO_START_DATA_REQ                           = 14,
  MCPM_DO_STOP_DATA_REQ                            = 15,
  MCPM_DO_PARMS_UPDATE_REQ                         = 16,

  /* GERAN Requests */
  MCPM_GERAN_START_REQ                             = 17,
  MCPM_GERAN_STOP_REQ                              = 18,
  MCPM_GERAN_WAKE_UP_REQ                           = 19,
  MCPM_GERAN_GO_TO_SLEEP_REQ                       = 20,
  MCPM_GERAN_IDLE_REQ                              = 21,
  MCPM_GERAN_VOICE_START_REQ                       = 22,
  MCPM_GERAN_VOICE_STOP_REQ                        = 23,
  MCPM_GERAN_DATA_START_REQ                        = 24,
  MCPM_GERAN_DATA_STOP_REQ                         = 25,
  MCPM_GERAN_PARMS_UPDATE_REQ                      = 26,

  /* GERAN1 Requests */
  MCPM_GERAN1_START_REQ                            = 27,
  MCPM_GERAN1_STOP_REQ                             = 28,
  MCPM_GERAN1_WAKE_UP_REQ                          = 29,
  MCPM_GERAN1_GO_TO_SLEEP_REQ                      = 30,
  MCPM_GERAN1_IDLE_REQ                             = 31,
```

```
MCPM_GERAN1_VOICE_START_REQ                          = 32,
MCPM_GERAN1_VOICE_STOP_REQ                           = 33,
MCPM_GERAN1_DATA_START_REQ                           = 34,
MCPM_GERAN1_DATA_STOP_REQ                            = 35,
MCPM_GERAN1_PARMS_UPDATE_REQ                         = 36,

/* LTE Requests */
MCPM_LTE_START_REQ                                   = 37,
MCPM_LTE_STOP_REQ                                    = 38,
MCPM_LTE_ACQ_REQ                                     = 39,
MCPM_LTE_WAKE_UP_REQ                                 = 40,
MCPM_LTE_GO_TO_SLEEP_REQ                             = 41,
MCPM_LTE_IDLE_REQ                                    = 42,
MCPM_LTE_DATA_START_REQ                              = 43,
MCPM_LTE_FDD_DATA_START_REQ                          = 44,
MCPM_LTE_TDD_DATA_START_REQ                          = 45,
MCPM_LTE_FDD_VOLTE_DATA_START_REQ                    = 46,
MCPM_LTE_TDD_VOLTE_DATA_START_REQ                    = 47,
MCPM_LTE_GO_TO_LIGHT_SLEEP_REQ                       = 48,
MCPM_LTE_GO_TO_LONG_LIGHT_SLEEP_REQ                  = 49,
MCPM_LTE_GO_TO_LIGHT_SLEEP_NO_MODEM_FREEZE_REQ       = 50,
MCPM_LTE_DATA_STOP_REQ                               = 51,
MCPM_LTE_PARMS_UPDATE_REQ                            = 52,

/* TDSCDMA Requests */
MCPM_TDSCDMA_START_REQ                               = 53,
MCPM_TDSCDMA_STOP_REQ                                = 54,
MCPM_TDSCDMA_ACQ_REQ                                 = 55,
MCPM_TDSCDMA_WAKE_UP_REQ                             = 56,
MCPM_TDSCDMA_GO_TO_SLEEP_REQ                         = 57,
MCPM_TDSCDMA_IDLE_REQ                                = 58,
MCPM_TDSCDMA_VOICE_START_REQ                         = 59,
MCPM_TDSCDMA_VOICE_STOP_REQ                          = 60,
MCPM_TDSCDMA_DATA_START_REQ                          = 61,
MCPM_TDSCDMA_DATA_STOP_REQ                           = 62,
MCPM_TDSCDMA_PARMS_UPDATE_REQ                        = 63,

/* WCDMA Requests */
MCPM_WCDMA_START_REQ                                 = 64,
MCPM_WCDMA_STOP_REQ                                  = 65,
MCPM_WCDMA_WAKE_UP_REQ                               = 66,
MCPM_WCDMA_GO_TO_SLEEP_REQ                           = 67,
MCPM_WCDMA_IDLE_REQ                                  = 68,
MCPM_WCDMA_VOICE_START_REQ                           = 69,
```

```
  MCPM_WCDMA_VOICE_STOP_REQ                    = 70,
  MCPM_WCDMA_DATA_START_REQ                    = 71,
  MCPM_WCDMA_DATA_STOP_REQ                     = 72,
  MCPM_WCDMA_PARMS_UPDATE_REQ                  = 73,

  /* GPS Requests */
  MCPM_GPS_STOP_REQ                            = 74,
  MCPM_GPS_ACQ_REQ                             = 75,
  MCPM_GPS_NON_DPO_REQ                         = 76,
  MCPM_GPS_DPO_ON_REQ                          = 77,
  MCPM_GPS_PARMS_UPDATE_REQ                    = 78,

/* RF Requests */
  MCPM_RF_START_REQ                            = 79,
  MCPM_RF_STOP_REQ                             = 80,

  /* A2 Requests */
  MCPM_A2_START_REQ                            = 81,
  MCPM_A2_STOP_REQ                             = 82,

  /* GSM Ciphering Requests */
  MCPM_GSM_CIPHERING_START_REQ                 = 83,
  MCPM_GSM_CIPHERING_STOP_REQ                  = 84,

  /* GSM Ciphering1 Requests */
  MCPM_GSM_CIPHERING1_START_REQ                = 85,
  MCPM_GSM_CIPHERING1_STOP_REQ                 = 86,

  MCPM_TECH_MAX_REQ
} mcpm_request_type;
```

3.2.2 Hexagon clock

Search for Sched resrc 9. In this example, LTE requested 307.2 MHz:

```
22:31:25.675:                    mcpm_npa.c:458      MCPM_NPA: Sched
resrc 9 req for 307200, tech 5
```

- Resrc – MCPM_Resrc_IDType
- Tech – Mcpm_tech_type

Type definition of MCPM_Resrc_IDType:

```
(MCPM_Resrc_IDType) enum(32 bits, signed, MCPM_RESRC_VDD_MSS_ID = 0,
                                          MCPM_RESRC_VDD_CX_ID = 1,
                                          MCPM_RESRC_VDD_MX_ID = 2,
                                          MCPM_RESRC_SNOC_ID = 3,
                                          MCPM_RESRC_BIMC_ID = 4,
                                          MCPM_RESRC_MSS_MODEM_BCR_ID = 5,
                                          MCPM_RESRC_MSS_AXI_MODEM_CBCR_ID
= 6,
                                          MCPM_RESRC_MSS_XO_MODEM_CBCR_ID =
7,
                                          MCPM_RESRC_MSS_XO_DTR_CBCR_ID =
8,
                                          MCPM_RESRC_CLK_Q6_ID = 9,
                                          MCPM_RESRC_CLK_PROC_ID = 10,
                                          MCPM_RESRC_CLK_TDEC_ID = 11,
                                          MCPM_RESRC_MODEM_MEM_PL_ENABLE_ID
= 12,
                                          MCPM_RESRC_MODEM_LMEM_ENABLE_ID =
13,
                                          MCPM_RESRC_MSS_STMR_BCR_ID = 14,
                                          MCPM_RESRC_MSS_BUS_STMR_CBCR_ID =
15,
                                          MCPM_RESRC_MSS_XO_STMR_CBCR_ID =
16,
                                          MCPM_RESRC_LATENCY_ID = 17,
                                          MCPM_MAX_RESRC_ID = 18,
                                          MCPM_NUM_RESRC = 18,
                                          MCPM_RESRC_ID_FORCE32BITS =
2147483647)
```

Mcpm_tech_type is defined in \modem_proc\mpower\api\mcpm_api.h and can differ from PL to PL.

```
typedef enum
{

  /* MCPM 1X technolgy definition */
  MCPM_1X_TECH,

  /* MCPM GERAN technolgy definition */
  MCPM_GERAN_TECH,

  /* MCPM GERAN1 technolgy definition */
```

```
  MCPM_GERAN1_TECH,

  /* MCPM DO technolgy definition */
  MCPM_DO_TECH,

  /* MCPM WCDMA technolgy definition */
  MCPM_WCDMA_TECH,

  /* MCPM LTE technolgy definition */
  MCPM_LTE_TECH,

  /* MCPM TDSCDMA technolgy definition */
  MCPM_TDSCDMA_TECH,

  /* MCPM GPS technolgy definition */
  MCPM_GPS_TECH,

  /* MCPM RF technology definition */
  MCPM_RF_TECH,

  /* MCPM A2 technolgy definition */
  MCPM_A2_TECH,

  /* MCPM GSM CIPHERING definition */
  MCPM_GSM_CIPHERING_TECH,

  /* MCPM GSM CIPHERING1 definition */
  MCPM_GSM_CIPHERING1_TECH,

  /* number fo techs used for boundary checks  */
  MCPM_NUM_TECH
}mcpm_tech_type;
```

The active Hexagon clock is in the NPA dump. In this example, the clock was the maximum clock 614.4 MHz since /node/core/cpu requested 729.6 MHz.

```
npa_resource (name: "/clk/cpu") (handle: 0x3C3E7A1C) (units: KHz) (resource
max: 614400) (active max: 0) (active state: 614400)  (active headroom:
614400) (request state: 729600)
 :         npa_client (name: wl1_npa_q6sw) (handle: 0x3C439568) (resource:
0x3C3E7A1C) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: gps_receiver) (handle: 0x3C6E3F68) (resource:
0x3C3E7A1C) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: GPS_MC_CPU_CLIENT) (handle: 0x3C6E4018)
(resource: 0x3C3E7A1C) (type: NPA_CLIENT_REQUIRED) (request: 0)
```

```
 :         npa_client (name: gps_pe) (handle: 0x3C6844D0) (resource:
0x3C3E7A1C) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: GPS_CC_CPU_CLIENT) (handle: 0x3C684528)
(resource: 0x3C3E7A1C) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: IPA_Q6_CPU_CLK_CLIENT) (handle: 0x3C4DC5B8)
(resource: 0x3C3E7A1C) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: a2_q6sw_cpu_clk_client) (handle: 0x3C4DC718)
(resource: 0x3C3E7A1C) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: mcpm_clk_q6) (handle: 0x3C4DCEC8) (resource:
0x3C3E7A1C) (type: NPA_CLIENT_SUPPRESSIBLE) (request: 307200)
 :         npa_client (name: timer_clk_client) (handle: 0x3C48A360)
(resource: 0x3C3E7A1C) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: npa_scheduler_clk_cpu_client) (handle:
0x3C439358) (resource: 0x3C3E7A1C) (type: NPA_CLIENT_SUPPRESSIBLE)
(request: 0)
 :         npa_client (name: /node/core/cpu) (handle: 0x3C439670)
(resource: 0x3C3E7A1C) (type: NPA_CLIENT_REQUIRED) (request: 729600)
 :         npa_client (name: /clk/cpu/impulse) (handle: 0x3C3E7F40)
(resource: 0x3C3E7A1C) (type: NPA_CLIENT_IMPULSE) (request: 0)
 :         npa_change_event (name: Clock_Change_Event) (handle: 0x3C42E488)
(resource: 0x3C3E7A1C)
 :         end npa_resource (handle: 0x3C3E7A1C)
```

3.2.3 CX, MX, MSS

Search for Sched resrc 0, Sched resrc 1, and Sched resrc 2 for MSS, CX, and MX. In this example, MCPM_VREG_NOM was requested for LTE.

```
22:31:25.675:                        mcpm_npa.c:458      MCPM_NPA: Sched
resrc 0 req for 4, tech 5
22:31:25.675:                        mcpm_npa.c:458      MCPM_NPA: Sched
resrc 1 req for 4, tech 5
22:31:25.675:                        mcpm_npa.c:458      MCPM_NPA: Sched
resrc 2 req for 4, tech 5
```

Type definition of MCPM_Voltage_DataType:

```
typedef enum
{
  MCPM_VREG_NO_VOTE   = 0,
  MCPM_VREG_RETENTION = 1,
  MCPM_VREG_SVS2      = 2,
  MCPM_VREG_SVS       = 3,
  MCPM_VREG_NOM       = 4,
  MCPM_VREG_NOM_PLUS  = 5,
```

```
  MCPM_VREG_TURBO     = 6,
  MCPM_VREG_MAX,
  MCPM_ENUM_32BITS(VREG_DATA)
} MCPM_Voltage_DataType;
```

3.2.4 Other resources

Search for Sched resrc *x*, where *x* is one of the resources in the following definition. These definitions are all the resources MCPM controls.

```
(MCPM_Resrc_IDType) enum(32 bits, signed, MCPM_RESRC_VDD_MSS_ID = 0,
                                          MCPM_RESRC_VDD_CX_ID = 1,
                                          MCPM_RESRC_VDD_MX_ID = 2,
                                          MCPM_RESRC_SNOC_ID = 3,
                                          MCPM_RESRC_BIMC_ID = 4,
                                          MCPM_RESRC_MSS_MODEM_BCR_ID = 5,
                                          MCPM_RESRC_MSS_AXI_MODEM_CBCR_ID
= 6,
                                          MCPM_RESRC_MSS_XO_MODEM_CBCR_ID =
7,
                                          MCPM_RESRC_MSS_XO_DTR_CBCR_ID =
8,
                                          MCPM_RESRC_CLK_Q6_ID = 9,
                                          MCPM_RESRC_CLK_PROC_ID = 10,
                                          MCPM_RESRC_CLK_TDEC_ID = 11,
                                          MCPM_RESRC_MODEM_MEM_PL_ENABLE_ID
= 12,
                                          MCPM_RESRC_MODEM_LMEM_ENABLE_ID =
13,
                                          MCPM_RESRC_MSS_STMR_BCR_ID = 14,
                                          MCPM_RESRC_MSS_BUS_STMR_CBCR_ID =
15,
                                          MCPM_RESRC_MSS_XO_STMR_CBCR_ID =
16,
                                          MCPM_RESRC_LATENCY_ID = 17,
                                          MCPM_MAX_RESRC_ID = 18,
                                          MCPM_NUM_RESRC = 18,
                                          MCPM_RESRC_ID_FORCE32BITS =
2147483647)
```

3.2.5 Matched system config IDs

Search for matched system config IDs. If there is no message like the one below, that means there was no MCPM configuration. This message should be printed between MCPM_Config_Modem: Start and MCPM_Config_Modem: End.

```
22:31:25.956:                          mcpm.c:1523     MCPM_Process_Req:
Matched system config IDs:   15   41    0, nbr bmask: 0x0 0x0 0x0 tick:
b9966c18, PC count q6:15109 rpm:15101
```

In the example above, the MCPM was configured for the following two types:

- MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_10M=15
- MCPM_CFG_A2_ON_DC_DC_DC_DC_DC_DC_DC_DC=41

```
typedef enum
{
  MCPM_CFG_POWER_DOWN_DC_DC_DC_DC_DC_DC_DC_DC=0,
  MCPM_CFG_LIGHT_SLEEP_DC_DC_DC_DC_DC_DC_DC_DC=1,
  MCPM_CFG_LIGHT_SLEEP_PC_DC_DC_DC_DC_DC_DC_DC_DC=2,
  MCPM_CFG_WCDMA_IDLE_RX_DC_DC_DC_DC_OFF_DC_DC_DC=3,
  MCPM_CFG_WCDMA_VOICE_DC_DC_DC_DC_OFF_DC_DC_DC=4,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS21_HS11_5M=5,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_ON_HS42_HS11_5M=6,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS42_HS11_10M=7,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS42_HS22_10M=8,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_ON_HS84_HS22_10M=9,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS63_HS22_15M=10,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS84_HS22_20M=11,
  MCPM_CFG_LTE_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_10M=12,
  MCPM_CFG_LTE_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_20M=13,
  MCPM_CFG_LTE_ACQ_DC_DC_DC_DC_DC_DC_DC_DC=14,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_10M=15,
  MCPM_CFG_LTE_VOLTE_DATA_DC_DC_DC_DC_DC_DC_DC_10M=16,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_20M=17,
  MCPM_CFG_LTE_VOLTE_DATA_DC_DC_DC_DC_DC_DC_DC_20M=18,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_5M_5M=19,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_10M_10M=20,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L300_L50_20M_20M=21,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L300_L100_20M_20M=22,
  MCPM_CFG_SVLTE_DC_DC_DC_DC_DC_L73_L23_10M=23,
  MCPM_CFG_C_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_NA=24,
  MCPM_CFG_C_STANDBY_DC_DC_DC_DC_DC_DC_DC_NA=25,
  MCPM_CFG_C_VOICE_DC_DC_DC_DC_DC_DC_DC_NA=26,
  MCPM_CFG_C_DATA_DC_DC_DC_DC_DC_DC_DC_NA=27,
```

```
  MCPM_CFG_DO_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC=28,
  MCPM_CFG_DO_DATA_DC_DC_DC_DC_DC_DO4_9_DO1_8_DC=29,
  MCPM_CFG_DO_DATA_DC_DC_DC_DC_DC_DO14_7_DO5_4_DC=30,
  MCPM_CFG_TDS_STANDBY_DC_DC_DC_DC_DC_DC_DC_DC=31,
  MCPM_CFG_TDS_ACQ_DC_DC_DC_DC_DC_DC_DC_DC=32,
  MCPM_CFG_TDS_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC=33,
  MCPM_CFG_TDS_VOICE_DC_DC_DC_DC_DC_DC_DC_DC=34,
  MCPM_CFG_TDS_DATA_DC_DC_DC_DC_DC_DC_DC_DC=35,
  MCPM_CFG_GSM_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC=36,
  MCPM_CFG_GSM_VOICE_DC_DC_DC_DC_DC_DC_DC_NA=37,
  MCPM_CFG_GSM_DATA_DC_DC_DC_DC_DC_NONEV_DC_NA=38,
  MCPM_CFG_GSM_DATA_DC_DC_DC_DC_DC_EV_EV_NA=39,
  MCPM_CFG_GSM_CIPHER_ON_DC_DC_DC_DC_DC_DC_DC_DC=40,
  MCPM_CFG_A2_ON_DC_DC_DC_DC_DC_DC_DC_DC=41,
  MCPM_CFG_GPS_ACQ_DC_DC_DC_DC_DC_DC_DC_DC=42,
  MCPM_CFG_RF_ON_DC_DC_DC_DC_DC_DC_DC_DC=43,
  /* Max sys cfg */
  MCPM_MAX_SYS_CFG,
  MCPM_NUM_SYS_CFG = MCPM_MAX_SYS_CFG,
  MCPM_ENUM_32BITS(SYS_CFG)
}MCPM_Sys_Cfg_IDType;
```


3.3 MSM8996

3.3.1 Check MCPM setting

- Locate MCPM_Config_Modem

```
22:31:25.956:                              mcpm.c:2377     Tech 5 --
MCPM_Config_Modem: Start of request: 81, begin time = 0x23b9966661,
pcycle = 0x8bc35b1c4f.  // MCPM_A2_START_REQ

22:31:25.956:                              mcpm.c:2403    Tech 5 --
MCPM_Config_Modem: End of request: 81, end time = 0x23b9969681, duration
= 641 uSec, pcycle = 0x8bc35b2564.  // MCPM_A2_START_REQ
```

  - Tech type – mcpm_tech_type
  - Request type– mcpm_request_type
  - Time – XOTimeTick
  - Pcycle – Hexagon pcycle
  - Duration – Configuration duration

The tech type and the request type are defined in \modem_proc\mpower\api\mcpm_api.h and can differ from PL to PL.

```
typedef enum
{

  /* MCPM 1X technology definition */
  MCPM_1X_TECH,

  /* MCPM DO technology definition */
  MCPM_DO_TECH,

  /* MCPM WCDMA technology definition */
  MCPM_WCDMA_TECH,

  /* MCPM LTE technology definition */
  MCPM_LTE_TECH,

  /* MCPM TDSCDMA technology definition */
  MCPM_TDSCDMA_TECH,

  /* MCPM GERAN technology definition */
  MCPM_GERAN_TECH,

  /* MCPM GERAN1 technology definition */
  MCPM_GERAN1_TECH,

  /* MCPM GPS technology definition */
  MCPM_GPS_TECH,

  /* MCPM RF technology definition */
  MCPM_RF_TECH,

  /* MCPM A2 technology definition */
  MCPM_A2_TECH,

  /* MCPM GSM CIPHERING definition */
  MCPM_GSM_CIPHERING_TECH,

  /* MCPM GSM CIPHERING1 definition */
  MCPM_GSM_CIPHERING1_TECH,

  /* number of techs used for boundary checks  */
  MCPM_NUM_TECH
```

```
}mcpm_tech_type;

typedef enum
{
  /* 1X Requests */
  MCPM_1X_START_REQ                                = 0,
  MCPM_1X_STOP_REQ                                 = 1,
  MCPM_1X_WAKE_UP_REQ                              = 2,
  MCPM_1X_GO_TO_SLEEP_REQ                          = 3,
  MCPM_1X_GO_TO_PSEUDO_SLEEP_REQ                   = 4,
  MCPM_1X_IDLE_REQ                                 = 5,
  MCPM_1X_VOICE_REQ                                = 6,
  MCPM_1X_DATA_REQ                                 = 7,
  MCPM_1X_PARMS_UPDATE_REQ                         = 8,

  /* DO Requests */
  MCPM_DO_START_REQ                                = 9,
  MCPM_DO_STOP_REQ                                 = 10,
  MCPM_DO_WAKE_UP_REQ                              = 11,
  MCPM_DO_GO_TO_SLEEP_REQ                          = 12,
  MCPM_DO_IDLE_REQ                                 = 13,
  MCPM_DO_START_DATA_REQ                           = 14,
  MCPM_DO_STOP_DATA_REQ                            = 15,
  MCPM_DO_PARMS_UPDATE_REQ                         = 16,

  /* GERAN Requests */
  MCPM_GERAN_START_REQ                             = 17,
  MCPM_GERAN_STOP_REQ                              = 18,
  MCPM_GERAN_WAKE_UP_REQ                           = 19,
  MCPM_GERAN_GO_TO_SLEEP_REQ                       = 20,
  MCPM_GERAN_IDLE_REQ                              = 21,
  MCPM_GERAN_VOICE_START_REQ                       = 22,
  MCPM_GERAN_VOICE_STOP_REQ                        = 23,
  MCPM_GERAN_DATA_START_REQ                        = 24,
  MCPM_GERAN_DATA_STOP_REQ                         = 25,
  MCPM_GERAN_PARMS_UPDATE_REQ                      = 26,

  /* GERAN1 Requests */
  MCPM_GERAN1_START_REQ                            = 27,
  MCPM_GERAN1_STOP_REQ                             = 28,
  MCPM_GERAN1_WAKE_UP_REQ                          = 29,
  MCPM_GERAN1_GO_TO_SLEEP_REQ                      = 30,
  MCPM_GERAN1_IDLE_REQ                             = 31,
  MCPM_GERAN1_VOICE_START_REQ                      = 32,
```

```
MCPM_GERAN1_VOICE_STOP_REQ                          = 33,
MCPM_GERAN1_DATA_START_REQ                          = 34,
MCPM_GERAN1_DATA_STOP_REQ                           = 35,
MCPM_GERAN1_PARMS_UPDATE_REQ                        = 36,

/* LTE Requests */
MCPM_LTE_START_REQ                                  = 37,
MCPM_LTE_STOP_REQ                                   = 38,
MCPM_LTE_INIT_REQ                                   = 39,
MCPM_LTE_ACQ_REQ                                    = 40,
MCPM_LTE_WAKE_UP_REQ                                = 41,
MCPM_LTE_GO_TO_SLEEP_REQ                            = 42,
MCPM_LTE_IDLE_REQ                                   = 43,
MCPM_LTE_DATA_START_REQ                             = 44,
MCPM_LTE_FDD_DATA_START_REQ                         = 45,
MCPM_LTE_TDD_DATA_START_REQ                         = 46,
MCPM_LTE_FDD_VOLTE_DATA_START_REQ                   = 47,
MCPM_LTE_TDD_VOLTE_DATA_START_REQ                   = 48,
MCPM_LTE_GO_TO_LIGHT_SLEEP_REQ                      = 49,
MCPM_LTE_GO_TO_LONG_LIGHT_SLEEP_REQ                 = 50,
MCPM_LTE_GO_TO_LIGHT_SLEEP_NO_MODEM_FREEZE_REQ      = 51,
MCPM_LTE_DATA_STOP_REQ                              = 52,
MCPM_LTE_PARMS_UPDATE_REQ                           = 53,

/* TDSCDMA Requests */
MCPM_TDSCDMA_START_REQ                              = 54,
MCPM_TDSCDMA_STOP_REQ                               = 55,
MCPM_TDSCDMA_ACQ_REQ                                = 56,
MCPM_TDSCDMA_WAKE_UP_REQ                            = 57,
MCPM_TDSCDMA_GO_TO_SLEEP_REQ                        = 58,
MCPM_TDSCDMA_IDLE_REQ                               = 59,
MCPM_TDSCDMA_VOICE_START_REQ                        = 60,
MCPM_TDSCDMA_VOICE_STOP_REQ                         = 61,
MCPM_TDSCDMA_DATA_START_REQ                         = 62,
MCPM_TDSCDMA_DATA_STOP_REQ                          = 63,
MCPM_TDSCDMA_PARMS_UPDATE_REQ                       = 64,

/* WCDMA Requests */
MCPM_WCDMA_START_REQ                                = 65,
MCPM_WCDMA_STOP_REQ                                 = 66,
MCPM_WCDMA_WAKE_UP_REQ                              = 67,
MCPM_WCDMA_GO_TO_SLEEP_REQ                          = 68,
MCPM_WCDMA_IDLE_REQ                                 = 69,
MCPM_WCDMA_VOICE_START_REQ                          = 70,
```

```
  MCPM_WCDMA_VOICE_STOP_REQ                          = 71,
  MCPM_WCDMA_DATA_START_REQ                          = 72,
  MCPM_WCDMA_CDRX_GO_TO_LIGHT_SLEEP_REQ              = 73,
  MCPM_WCDMA_DATA_STOP_REQ                           = 74,
  MCPM_WCDMA_PARMS_UPDATE_REQ                        = 75,

  /* GPS Requests */
  MCPM_GPS_STOP_REQ                                  = 76,
  MCPM_GPS_ACQ_REQ                                   = 77,
  MCPM_GPS_NON_DPO_REQ                               = 78,
  MCPM_GPS_DPO_ON_REQ                                = 79,
  MCPM_GPS_PARMS_UPDATE_REQ                          = 80,

/* RF Requests */
  MCPM_RF_START_REQ                                  = 81,
  MCPM_RF_STOP_REQ                                   = 82,

  /* A2 Requests */
  MCPM_A2_START_REQ                                  = 83,
  MCPM_A2_STOP_REQ                                   = 84,

  /* GSM Ciphering Requests */
  MCPM_GSM_CIPHERING_START_REQ                       = 85,
  MCPM_GSM_CIPHERING_STOP_REQ                        = 86,

  /* GSM Ciphering1 Requests */
  MCPM_GSM_CIPHERING1_START_REQ                      = 87,
  MCPM_GSM_CIPHERING1_STOP_REQ                       = 88,

  MCPM_TECH_MAX_REQ
} mcpm_request_type;
```

3.3.2 Hexagon clock

Locate the Sched resrc 9. In the following example, LTE requests 307.2 MHz.

```
22:31:25.675:                          mcpm_npa.c:458      MCPM_NPA: Sched
resrc 9 req for 307200, tech 5
```

- Resrc – MCPM_Resrc_IDType
- Tech – Mcpm_tech_type

The type definition of MCPM_Resrc_IDType is as follows:

```
type (MCPM_Resrc_IDType)
enum(32 bits, signed,
MCPM_RESRC_MCA_ID = 0,
MCPM_RESRC_VDD_MSS_ID = 1,
MCPM_RESRC_VDD_CX_ID = 2,
MCPM_RESRC_VDD_MX_ID = 3,
MCPM_RESRC_IPA_SNOC_ID = 4,
MCPM_RESRC_MODEM_BIMC_ID = 5,
MCPM_RESRC_Q6_BIMC_ID = 6,
MCPM_RESRC_MSS_MODEM_BCR_ID = 7,
MCPM_RESRC_MSS_BUS_MGPI_CBCR_ID = 8,
MCPM_RESRC_CLK_Q6_ID = 9,
MCPM_RESRC_CLK_PROC_ID = 10,
MCPM_RESRC_CLK_TDEC_ID = 11,
MCPM_RESRC_MCDMA_BIMC_CLK_ID = 12,
MCPM_RESRC_MSS_BUS_STMR_CBCR_ID = 13,
MCPM_RESRC_LATENCY_ID = 14,
MCPM_RESRC_FW_RFCCS_ID = 15,
MCPM_RESRC_CORE_CPU_VDD_ID = 16,
MCPM_MAX_RESRC_ID = 17,
MCPM_NUM_RESRC = 17,
MCPM_RESRC_ID_FORCE32BITS = 2147483647)
```

The active Hexagon clock is in the NPA dump. In the following example, a maximum clock of 844.8 MHz was used as the /node/core/cpu requested 729.6 MHz.

```
: npa_resource (name: "/clk/cpu") (handle: 0xAF4830A8) (units: KHz)
(resource max: 844800) (active max: 844800) (active state: 844800)  (active
headroom: 0) (request state: 844800)
: npa_client (name: gps_pe) (handle: 0xAF728A28) (resource: 0xAF4830A8)
(type: NPA_CLIENT_REQUIRED) (request: 0)
: npa_client (name: gps_rx) (handle: 0xAF728D18) (resource: 0xAF4830A8)
(type: NPA_CLIENT_REQUIRED) (request: 0)
```

```
: npa_client (name: GPS_MC_CPU_CLIENT) (handle: 0xAF728DB8) (resource:
0xAF4830A8) (type: NPA_CLIENT_REQUIRED) (request: 0)
: npa_client (name: GPS_CC_CPU_CLIENT) (handle: 0xAF728EA8) (resource:
0xAF4830A8) (type: NPA_CLIENT_REQUIRED) (request: 0)
: npa_client (name: RFLM_W_TX) (handle: 0xAF728F98) (resource: 0xAF4830A8)
(type: NPA_CLIENT_REQUIRED) (request: 0)
: npa_client (name: a2_q6sw_cpu_clk_client) (handle: 0xAF57E578) (resource:
0xAF4830A8) (type: NPA_CLIENT_REQUIRED) (request: 0)
: npa_client (name: wci2_dcvs) (handle: 0xAF57E668) (resource: 0xAF4830A8)
(type: NPA_CLIENT_REQUIRED) (request: 0)
: npa_client (name: mcpm_fw_cpu_boost) (handle: 0xAF57E6B8) (resource:
0xAF4830A8) (type: NPA_CLIENT_REQUIRED) (request: 0)
: npa_client (name: timer_clk_client) (handle: 0xAF50A2E8) (resource:
0xAF4830A8) (type: NPA_CLIENT_REQUIRED) (request: 0)
: npa_client (name: npa_scheduler_clk_cpu_client) (handle: 0xAF496B38)
(resource: 0xAF4830A8) (type: NPA_CLIENT_SUPPRESSIBLE) (request: 0)
: npa_client (name: /node/core/mca) (handle: 0xAF496158) (resource:
0xAF4830A8) (type: 32768) (request: 0)
: npa_client (name: /node/core/mca) (handle: 0xAF4961A8) (resource:
0xAF4830A8) (type: NPA_CLIENT_REQUIRED) (request: 0)
: npa_client (name: /node/core/cpu) (handle: 0xAF4963D8) (resource:
0xAF4830A8) (type: NPA_CLIENT_REQUIRED) (request: 844800)
: npa_client (name: /clk/cpu/impulse) (handle: 0xAF476138) (resource:
0xAF4830A8) (type: NPA_CLIENT_IMPULSE) (request: 0)
: npa_reserved_event (name: ) (handle: 0xAF43E8C8) (resource: 0xAF4830A8)
: end npa_resource (handle: 0xAF4830A8)
```

3.3.3 CX, MX, MSS

Locate the Sched resrc 1, Sched resrc 2, and Sched resrc 3 for MSS, CX, and MX. In the following example, MCPM_VREG_NOM is requested for LTE.

```
22:31:25.675:                    mcpm_npa.c:458      MCPM_NPA: Sched
resrc 1 req for 4, tech 5
22:31:25.675:                    mcpm_npa.c:458      MCPM_NPA: Sched
resrc 2 req for 4, tech 5
22:31:25.675:                    mcpm_npa.c:458      MCPM_NPA: Sched
resrc 3 req for 4, tech 5
```

The type definition of MCPM_Voltage_DataType is as follows:

```
type (MCPM_Voltage_DataType)
enum(32 bits, signed,
MCPM_VREG_NO_VOTE = 0,
MCPM_VREG_RETENTION = 1,
MCPM_VREG_SVS2 = 2,
```

```
MCPM_VREG_SVS = 3,
MCPM_VREG_NOM = 4,
MCPM_VREG_NOM_PLUS = 5,
MCPM_VREG_TURBO = 6,
MCPM_VREG_MAX = 7,
MCPM_VREG_DATA_FORCE32BITS = 2147483647)
```

3.3.4 Other resources

Locate the Sched resrc *x*, where *x* is one of the resources in the following definition. The definitions are the resources MCPM controls.

```
type (MCPM_Resrc_IDType)
enum(32 bits, signed,
MCPM_RESRC_MCA_ID = 0,
MCPM_RESRC_VDD_MSS_ID = 1,
MCPM_RESRC_VDD_CX_ID = 2,
MCPM_RESRC_VDD_MX_ID = 3,
MCPM_RESRC_IPA_SNOC_ID = 4,
MCPM_RESRC_MODEM_BIMC_ID = 5,
MCPM_RESRC_Q6_BIMC_ID = 6,
MCPM_RESRC_MSS_MODEM_BCR_ID = 7,
MCPM_RESRC_MSS_BUS_MGPI_CBCR_ID = 8,
MCPM_RESRC_CLK_Q6_ID = 9,
MCPM_RESRC_CLK_PROC_ID = 10,
MCPM_RESRC_CLK_TDEC_ID = 11,
MCPM_RESRC_MCDMA_BIMC_CLK_ID = 12,
MCPM_RESRC_MSS_BUS_STMR_CBCR_ID = 13,
MCPM_RESRC_LATENCY_ID = 14,
MCPM_RESRC_FW_RFCCS_ID = 15,
MCPM_RESRC_CORE_CPU_VDD_ID = 16,
MCPM_MAX_RESRC_ID = 17,
MCPM_NUM_RESRC = 17,
MCPM_RESRC_ID_FORCE32BITS = 2147483647)
```

3.3.5 Matched system config IDs

Locate the matched system config IDs by checking for the following message. The message is printed between MCPM_Config_Modem: Start and MCPM_Config_Modem: End.

NOTE: If no such message is found then MCPM is not configured.

```
22:31:25.956:                              mcpm.c:1523     MCPM_Process_Req:
Matched system config IDs:   18   47    0, nbr bmask: 0x0 0x0 0x0 tick:
b9966c18, PC count q6:15109 rpm:15101
```

The MCPM is configured as follows:

- MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_10M=18
- MCPM_CFG_A2_ON_DC_DC_DC_DC_DC_DC_DC_DC=47

```
typedef enum
{
  MCPM_CFG_POWER_DOWN_DC_DC_DC_DC_DC_DC_DC_DC=0,
  MCPM_CFG_SLEEP_DC_DC_DC_DC_DC_DC_DC_DC=1,
  MCPM_CFG_LIGHT_SLEEP_DC_DC_DC_DC_DC_DC_DC_DC=2,
  MCPM_CFG_LIGHT_SLEEP_PC_DC_DC_DC_DC_DC_DC_DC=3,
  MCPM_CFG_W_LIGHT_SLEEP_DC_DC_DC_DC_DC_DC_DC_DC=4,
  MCPM_CFG_WCDMA_IDLE_RX_DC_DC_DC_DC_OFF_DC_DC_DC=5,
  MCPM_CFG_WCDMA_VOICE_DC_DC_DC_DC_OFF_DC_DC_DC=6,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS21_HS11_5M=7,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_ON_HS42_HS11_5M=8,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS42_HS11_10M=9,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS42_HS22_10M=10,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_ON_HS84_HS22_10M=11,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS63_HS22_15M=12,
  MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS84_HS22_20M=13,
  MCPM_CFG_LTE_INIT_DC_DC_DC_DC_DC_DC_DC_DC=14,
  MCPM_CFG_LTE_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_10M=15,
  MCPM_CFG_LTE_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_20M=16,
  MCPM_CFG_LTE_ACQ_DC_DC_DC_DC_DC_DC_DC_DC=17,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_10M=18,
  MCPM_CFG_LTE_VOLTE_DATA_DC_DC_DC_DC_DC_DC_DC_10M=19,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_20M=20,
  MCPM_CFG_LTE_VOLTE_DATA_DC_DC_DC_DC_DC_DC_DC_20M=21,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_5M_5M=22,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_DC_10M_10M=23,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L400_L75_20M_20M=24,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L400_L150_20M_20M=25,
  MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L200_L37_10M_5M_5M=26,
```

```
MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L450_L50_20M_20M_20M=27,
MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L600_L75_20M_20M_20M=28,
MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_DC_L150_20M_20M_20M=29,
MCPM_CFG_C_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_NA=30,
MCPM_CFG_C_STANDBY_DC_DC_DC_DC_DC_DC_DC_NA=31,
MCPM_CFG_C_VOICE_DC_DC_DC_DC_DC_DC_DC_NA=32,
MCPM_CFG_C_DATA_DC_DC_DC_DC_DC_DC_DC_NA=33,
MCPM_CFG_DO_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC=34,
MCPM_CFG_DO_DATA_DC_DC_DC_DC_DC_DO4_9_DO1_8_DC=35,
MCPM_CFG_DO_DATA_DC_DC_DC_DC_DC_DO14_7_DO5_4_DC=36,
MCPM_CFG_TDS_STANDBY_DC_DC_DC_DC_DC_DC_DC_DC=37,
MCPM_CFG_TDS_ACQ_DC_DC_DC_DC_DC_DC_DC_DC=38,
MCPM_CFG_TDS_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC=39,
MCPM_CFG_TDS_VOICE_DC_DC_DC_DC_DC_DC_DC_DC=40,
MCPM_CFG_TDS_DATA_DC_DC_DC_DC_DC_DC_DC_DC=41,
MCPM_CFG_GSM_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC=42,
MCPM_CFG_GSM_VOICE_DC_DC_DC_DC_DC_DC_DC_NA=43,
MCPM_CFG_GSM_DATA_DC_DC_DC_DC_DC_NONEV_DC_NA=44,
MCPM_CFG_GSM_DATA_DC_DC_DC_DC_DC_EV_EV_NA=45,
MCPM_CFG_GSM_CIPHER_ON_DC_DC_DC_DC_DC_DC_DC_DC=46,
MCPM_CFG_A2_ON_DC_DC_DC_DC_DC_DC_DC_DC=47,
MCPM_CFG_GPS_ACQ_DC_DC_DC_DC_DC_DC_DC_DC=48,
MCPM_CFG_RF_ON_DC_DC_DC_DC_DC_DC_DC_DC=49,
/* Max sys cfg */
MCPM_MAX_SYS_CFG,
MCPM_NUM_SYS_CFG = MCPM_MAX_SYS_CFG,
MCPM_ENUM_32BITS(SYS_CFG)
 }MCPM_Sys_Cfg_IDType;
```

3.4 MSM8952

NOTE: This section was added to this document revision.

3.4.1 Check MCPM setting

- Locate MCPM_Config_Modem

```
19:25:55.177:                              mcpm.c:3023
MCPM_Config_Modem: Start of request: 52, begin time = 0x36dbcb6d3f,
pcycle 0x8bc38873d7.  // MCPM_LTE_GO_TO_SLEEEP_REQ

19:25:55.177:                              mcpm.c:3052
MCPM_Config_Modem: End of request: 52, end time = 0x36dbcb9681, duration
= 550 uSec, pcycle = 0x8bc38badc8.  // MCPM_LTE_GO_TO_SLEEEP_REQ
```

  - Request type– mcpm_request_type
  - Time – XOTimeTick
  - Pcycle – Hexagon pcycle
  - Duration – Configuration duration

The technology type and the request type are defined in \modem_proc\mpower\api\mcpm_api.h and can differ from PL to PL.

```
typedef enum
{

  /* MCPM 1X technolgy definition */
  MCPM_1X_TECH,

  /* MCPM GERAN technolgy definition */
  MCPM_GERAN_TECH,

 /* MCPM GERAN1 technolgy definition */
  MCPM_GERAN1_TECH,

   /* MCPM GERAN2 technolgy definition */
  MCPM_GERAN2_TECH,

  /* MCPM DO technolgy definition */
  MCPM_DO_TECH,

  /* MCPM WCDMA technolgy definition */
  MCPM_WCDMA_TECH,
```

```
  /* MCPM LTE technolgy definition */
  MCPM_LTE_TECH,

  /* MCPM TDSCDMA technolgy definition */
  MCPM_TDSCDMA_TECH,

  /* MCPM GPS technolgy definition */
  MCPM_GPS_TECH,

  /* MCPM RF technology definition */
  MCPM_RF_TECH,

  /* MCPM GSM CIPHERING definition */
  MCPM_GSM_CIPHERING_TECH,

  /* MCPM GSM1 CIPHERING definition */
  MCPM_GSM1_CIPHERING_TECH,

    /* MCPM GSM2 CIPHERING definition */
  MCPM_GSM2_CIPHERING_TECH,

  /* MCPM A2 technolgy definition */
  MCPM_A2_TECH,

  /* number fo techs used for boundary checks  */
  MCPM_NUM_TECH
} mcpm_tech_type;

typedef enum
{
  /* 1X Requests */
  MCPM_1X_START_REQ              = 0,
  MCPM_1X_STOP_REQ               = 1,
  MCPM_1X_WAKE_UP_REQ            = 2,
  MCPM_1X_GO_TO_SLEEP_REQ        = 3,
  MCPM_1X_GO_TO_PSEUDO_SLEEP_REQ = 4,
  MCPM_1X_IDLE_REQ               = 5,
  MCPM_1X_VOICE_REQ              = 6,
  MCPM_1X_DATA_REQ               = 7,
  MCPM_1X_PARMS_UPDATE_REQ       = 8,

  /* DO Requests */
  MCPM_DO_START_REQ              = 9,
  MCPM_DO_STOP_REQ               = 10,
```

```
MCPM_DO_WAKE_UP_REQ            = 11,
MCPM_DO_GO_TO_SLEEP_REQ        = 12,
MCPM_DO_IDLE_REQ               = 13,
MCPM_DO_START_DATA_REQ         = 14,
MCPM_DO_STOP_DATA_REQ          = 15,
MCPM_DO_PARMS_UPDATE_REQ       = 16,

/* GERAN Requests */
MCPM_GERAN_START_REQ           = 17,
MCPM_GERAN_STOP_REQ            = 18,
MCPM_GERAN_WAKE_UP_REQ         = 19,
MCPM_GERAN_GO_TO_SLEEP_REQ     = 20,
MCPM_GERAN_IDLE_REQ            = 21,
MCPM_GERAN_VOICE_START_REQ     = 22,
MCPM_GERAN_VOICE_STOP_REQ      = 23,
MCPM_GERAN_DATA_START_REQ      = 24,
MCPM_GERAN_DATA_STOP_REQ       = 25,
MCPM_GERAN_PARMS_UPDATE_REQ    = 26,

/* GERAN1 Requests */
MCPM_GERAN1_START_REQ                   = 27,
MCPM_GERAN1_STOP_REQ                    = 28,
MCPM_GERAN1_WAKE_UP_REQ                 = 29,
MCPM_GERAN1_GO_TO_SLEEP_REQ             = 30,
MCPM_GERAN1_IDLE_REQ                    = 31,
MCPM_GERAN1_VOICE_START_REQ             = 32,
MCPM_GERAN1_VOICE_STOP_REQ              = 33,
MCPM_GERAN1_DATA_START_REQ              = 34,
MCPM_GERAN1_DATA_STOP_REQ               = 35,
MCPM_GERAN1_PARMS_UPDATE_REQ            = 36,

/* GERAN2 Requests */
MCPM_GERAN2_START_REQ                   = 37,
MCPM_GERAN2_STOP_REQ                    = 38,
MCPM_GERAN2_WAKE_UP_REQ                 = 39,
MCPM_GERAN2_GO_TO_SLEEP_REQ             = 40,
MCPM_GERAN2_IDLE_REQ                    = 41,
MCPM_GERAN2_VOICE_START_REQ             = 42,
MCPM_GERAN2_VOICE_STOP_REQ              = 43,
MCPM_GERAN2_DATA_START_REQ              = 44,
MCPM_GERAN2_DATA_STOP_REQ               = 45,
MCPM_GERAN2_PARMS_UPDATE_REQ            = 46,

/* LTE Requests */
```

```
  MCPM_LTE_START_REQ                                  = 47,
  MCPM_LTE_STOP_REQ                                   = 48,
  MCPM_LTE_INIT_REQ                                   = 49,
  MCPM_LTE_ACQ_REQ                                    = 50,
  MCPM_LTE_WAKE_UP_REQ                                = 51,
  MCPM_LTE_GO_TO_SLEEP_REQ                            = 52,
  MCPM_LTE_IDLE_REQ                                   = 53,
  MCPM_LTE_TDD_DATA_START_REQ                         = 54,
  MCPM_LTE_FDD_DATA_START_REQ                         = 55,
  MCPM_LTE_TDD_VOLTE_DATA_START_REQ                   = 56,
  MCPM_LTE_FDD_VOLTE_DATA_START_REQ                   = 57,
  /* To maintain backwards compatibility map data start to fdd data
start - going forward LTE Data
   * has either FDD or TDD only state so the Data_Start is redundanct
   */
  MCPM_LTE_DATA_START_REQ                             =
MCPM_LTE_FDD_DATA_START_REQ,
  MCPM_LTE_GO_TO_LIGHT_SLEEP_REQ                      = 58,
  MCPM_LTE_GO_TO_LONG_LIGHT_SLEEP_REQ                 = 59,
  MCPM_LTE_GO_TO_LIGHT_SLEEP_NO_MODEM_FREEZE_REQ      = 60,
  MCPM_LTE_DATA_STOP_REQ                              = 61,
  MCPM_LTE_PARMS_UPDATE_REQ                           = 62,

  /* TDSCDMA Requests */
  MCPM_TDSCDMA_START_REQ                              = 63,
  MCPM_TDSCDMA_STOP_REQ                               = 64,
  MCPM_TDSCDMA_ACQ_REQ                                = 65,
  MCPM_TDSCDMA_WAKE_UP_REQ                            = 66,
  MCPM_TDSCDMA_GO_TO_SLEEP_REQ                        = 67,
  MCPM_TDSCDMA_IDLE_REQ                               = 68,
  MCPM_TDSCDMA_VOICE_START_REQ                        = 69,
  MCPM_TDSCDMA_VOICE_STOP_REQ                         = 70,
  MCPM_TDSCDMA_DATA_START_REQ                         = 71,
  MCPM_TDSCDMA_DATA_STOP_REQ                          = 72,
  MCPM_TDSCDMA_PARMS_UPDATE_REQ                       = 73,

  /* WCDMA Requests */
  MCPM_WCDMA_START_REQ                                = 74,
  MCPM_WCDMA_STOP_REQ                                 = 75,
  MCPM_WCDMA_WAKE_UP_REQ                              = 76,
  MCPM_WCDMA_GO_TO_SLEEP_REQ                          = 77,
  MCPM_WCDMA_IDLE_REQ                                 = 78,
  MCPM_WCDMA_CDRX_GO_TO_LIGHT_SLEEP_REQ               = 79,
  MCPM_WCDMA_VOICE_START_REQ                          = 80,
  MCPM_WCDMA_VOICE_STOP_REQ                           = 81,
```

```
  MCPM_WCDMA_DATA_START_REQ                          = 82,
  MCPM_WCDMA_DATA_STOP_REQ                           = 83,
  MCPM_WCDMA_PARMS_UPDATE_REQ                        = 84,

  /* GPS Requests */
  MCPM_GPS_STOP_REQ                                  = 85,
  MCPM_GPS_ACQ_REQ                                   = 86,
  MCPM_GPS_NON_DPO_REQ                               = 87,
  MCPM_GPS_DPO_ON_REQ                                = 88,
  MCPM_GPS_PARMS_UPDATE_REQ                          = 89,

/* RF Requests */
  MCPM_RF_START_REQ                                  = 90,
  MCPM_RF_STOP_REQ                                   = 91,

  /* A2 Requests */
  MCPM_A2_START_REQ                                  = 92,
  MCPM_A2_STOP_REQ                                   = 93,

  /* GSM CIPHERING Requests */
  MCPM_GSM_CIPHERING_START_REQ                       = 94,
  MCPM_GSM_CIPHERING_STOP_REQ                        = 95,

    /* GSM1 CIPHERING Requests */
  MCPM_GSM_CIPHERING1_START_REQ                      = 96,
  MCPM_GSM_CIPHERING1_STOP_REQ                       = 97,

      /* GSM2 CIPHERING Requests */
  MCPM_GSM_CIPHERING2_START_REQ                      = 98,
  MCPM_GSM_CIPHERING2_STOP_REQ                       = 99,

  MCPM_TECH_MAX_REQ
}
mcpm_request_type;
```

3.4.2 Hexagon clock

Locate the Sched resrc 15. In the following example, LTE requests 576 MHz:

```
22:31:25.675:                    mcpm_npa.c:458      MCPM_NPA: Sched
resrc 15 req for 576000, tech 5
```

- Resrc – MCPM_Resrc_IDType
- Tech – mcpm_tech_type

The type definition of MCPM_Resrc_IDType is as follows:

```
type(MCPM_Resrc_IDType)
enum(32 bits, signed,
MCPM_RESRC_VDD_CX_ID = 0,
MCPM_RESRC_VDD_MX_ID = 1,
MCPM_RESRC_VDD_MSS_ID = 2,
MCPM_RESRC_MSS_ENABLE_ID = 3,
MCPM_RESRC_MTC_ENABLE_ID = 4,
MCPM_RESRC_DEMBACK_ENABLE_ID = 5,
MCPM_RESRC_TDEC_CLK_CTL_ID = 6,
MCPM_RESRC_TDEC_CLK_CTL_DB1_ID = 7,
MCPM_RESRC_MODEM_PWRUP_ID = 8,
MCPM_RESRC_UNIV_STMR_ENABLE_ID = 9,
MCPM_RESRC_MSS_MPLL1_ID = 10,
MCPM_RESRC_MSS_BUS_MGPI_CBCR_ID = 11,
MCPM_RESRC_FW_RFCCS_ID = 12,
MCPM_RESRC_MEM_SLP_CNTL_ID = 13,
MCPM_RESRC_MCA_ID = 14,
MCPM_RESRC_CLK_Q6_ID = 15,
MCPM_RESRC_CLK_Q6_CP_ID = 16,
MCPM_RESRC_CLK_BUS_ID = 17,
MCPM_RESRC_CLK_MODEM_MSSBUS_ID = 18,
MCPM_RESRC_CLK_MODEM_MTC_FAST_ID = 19,
MCPM_RESRC_CLK_MODEM_AXI_ID = 20,
MCPM_RESRC_CLK_MODEM_CCS_ID = 21,
MCPM_RESRC_CLK_TDEC_ID = 22,
MCPM_RESRC_CLK_SNOC_ID = 23,
MCPM_RESRC_CLK_BIMC_ID = 24,
MCPM_RESRC_LATENCY_ID = 25,
MCPM_RESRC_DISABLE_Q6_DLS_ID = 26,
MCPM_MAX_RESRC_ID = 27,
MCPM_NUM_RESRC = 27,
MCPM_RESRC_ID_FORCE32BITS = 2147483647)
```

The active Hexagon clock is in the NPA dump. In the following example, a maximum clock of 691.2 MHz was used as the /node/core/cpu requested 576 MHz:

```
: npa_resource (name: "/clk/cpu") (handle: 0x8AA95764) (units: KHz)
(resource max: 691200) (active max: 691200) (active state: 576000)
(active headroom: -115200) (request state: 576000)
 :         npa_client (name: wl1_npa_q6sw) (handle: 0x8AAA3AC8)
(resource: 0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: gps_rx) (handle: 0x8ABDBA88) (resource:
0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: GPS_MC_CPU_CLIENT) (handle: 0x8ABDBB78)
(resource: 0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: gps_pe) (handle: 0x8ABDBC68) (resource:
0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: GPS_CC_CPU_CLIENT) (handle: 0x8AB892E8)
(resource: 0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: RFLM_W_TX) (handle: 0x8AB893D8) (resource:
0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: IPA_Q6_CPU_CLK_CLIENT) (handle: 0x8AB103E8)
(resource: 0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: a2_q6sw_cpu_clk_client) (handle:
0x8AB10578) (resource: 0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request:
0)
 :         npa_client (name: mcpm_cpu_boost_client) (handle: 0x8AB10668)
(resource: 0x8AA95764) (type: NPA_CLIENT_SUPPRESSIBLE) (request: 0)
 :         npa_client (name: mcpm_nrat_q6freq_client) (handle:
0x8AB10148) (resource: 0x8AA95764) (type: NPA_CLIENT_SUPPRESSIBLE)
(request: 576000)
 :         npa_client (name: mcpm_fw_cpu_boost) (handle: 0x8AB10238)
(resource: 0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: timer_clk_client) (handle: 0x8AAEAFC8)
(resource: 0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 0)
 :         npa_client (name: npa_scheduler_clk_cpu_client) (handle:
0x8AAA3848) (resource: 0x8AA95764) (type: NPA_CLIENT_SUPPRESSIBLE)
(request: 0)
 :         npa_client (name: /node/core/mca) (handle: 0x8AAA3A28)
(resource: 0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 543312)
 :         npa_client (name: /node/core/cpu) (handle: 0x8AAA1A98)
(resource: 0x8AA95764) (type: NPA_CLIENT_REQUIRED) (request: 288000)
 :         npa_client (name: /clk/cpu/impulse) (handle: 0x8AA96A38)
(resource: 0x8AA95764) (type: NPA_CLIENT_IMPULSE) (request: 0)
 :         npa_reserved_event (name: ) (handle: 0x8AA67780) (resource:
0x8AA95764)
 :         end npa_resource (handle: 0x8AA95764)
```

3.4.3 CX, MX, MSS

Locate the Sched resrc 0, Sched resrc 1, and Sched resrc 2 for CX, MX and MSS. In the following example, MCPM_VREG_NOM is requested for LTE.

```
22:31:25.675:                    mcpm_npa.c:458      MCPM_NPA: Sched
resrc 0 req for 5, tech 5
22:31:25.675:                    mcpm_npa.c:458      MCPM_NPA: Sched
resrc 1 req for 5, tech 5
22:31:25.675:                    mcpm_npa.c:458      MCPM_NPA: Sched
resrc 2 req for 5, tech 5
```

The type definition of MCPM_Voltage_DataType is as follows:

```
Type (MCPM_Voltage_DataType)
enum(32 bits, signed,
MCPM_VREG_NO_VOTE = 0,
MCPM_VREG_RETENTION = 1,
MCPM_VREG_SVS_MINUS = 2,
MCPM_VREG_SVS = 3,
MCPM_VREG_SVS_PLUS = 4,
MCPM_VREG_NOM = 5,
MCPM_VREG_NOM_PLUS = 6,
MCPM_VREG_TURBO = 7,
MCPM_VREG_MAX = 8,
MCPM_VREG_DATA_FORCE32BITS = 2147483647)
```

3.4.4 Other resources

Locate the Sched resrc *x*, where *x* is one of the resources in the following definition. The definitions are the resources MCPM controls.

```
type(MCPM_Resrc_IDType)
enum(32 bits, signed,
MCPM_RESRC_VDD_CX_ID = 0,
MCPM_RESRC_VDD_MX_ID = 1,
MCPM_RESRC_VDD_MSS_ID = 2,
MCPM_RESRC_MSS_ENABLE_ID = 3,
MCPM_RESRC_MTC_ENABLE_ID = 4,
MCPM_RESRC_DEMBACK_ENABLE_ID = 5,
MCPM_RESRC_TDEC_CLK_CTL_ID = 6,
MCPM_RESRC_TDEC_CLK_CTL_DB1_ID = 7,
MCPM_RESRC_MODEM_PWRUP_ID = 8,
MCPM_RESRC_UNIV_STMR_ENABLE_ID = 9,
MCPM_RESRC_MSS_MPLL1_ID = 10,
MCPM_RESRC_MSS_BUS_MGPI_CBCR_ID = 11,
MCPM_RESRC_FW_RFCCS_ID = 12,
```

```
MCPM_RESRC_MEM_SLP_CNTL_ID = 13,
MCPM_RESRC_MCA_ID = 14,
MCPM_RESRC_CLK_Q6_ID = 15,
MCPM_RESRC_CLK_Q6_CP_ID = 16,
MCPM_RESRC_CLK_BUS_ID = 17,
MCPM_RESRC_CLK_MODEM_MSSBUS_ID = 18,
MCPM_RESRC_CLK_MODEM_MTC_FAST_ID = 19,
MCPM_RESRC_CLK_MODEM_AXI_ID = 20,
MCPM_RESRC_CLK_MODEM_CCS_ID = 21,
MCPM_RESRC_CLK_TDEC_ID = 22,
MCPM_RESRC_CLK_SNOC_ID = 23,
MCPM_RESRC_CLK_BIMC_ID = 24,
MCPM_RESRC_LATENCY_ID = 25,
MCPM_RESRC_DISABLE_Q6_DLS_ID = 26,
MCPM_MAX_RESRC_ID = 27,
MCPM_NUM_RESRC = 27,
MCPM_RESRC_ID_FORCE32BITS = 2147483647)
```

3.4.5 Matched system config IDs

Locate the matched system config IDs by checking the following message. The message is printed between MCPM_Config_Modem: Start and MCPM_Config_Modem: End.

NOTE: If no such message is found, then MCPM is not configured.

```
19:25:48.558:  mcpm.c:2124     MCPM_Process_Req: Matched system config IDs:
27   45    0, nbr bmask: 0x0 0x0 0x0 tick: 0xd4380643, PC count q6:77444
rpm:34535
```

The MCPM is configured as follows:

- MCPM_CFG_LTE_FDD_DATA_DC_DC_DC_DC_DC_L73_L23_10M = 27
- MCPM_CFG_A2_ON_DC_DC_DC_DC_DC_DC_DC_DC = 45

```
type(MCPM_Sys_Cfg_IDType)
enum(32 bits, signed,
MCPM_CFG_POWER_DOWN_DC_DC_DC_DC_DC_DC_DC_DC = 0,
MCPM_CFG_SLEEP_DC_DC_DC_DC_DC_DC_DC_DC = 1,
MCPM_CFG_LIGHT_SLEEP_DC_DC_DC_DC_DC_DC_DC_DC = 2,
MCPM_CFG_LIGHT_SLEEP_PC_DC_DC_DC_DC_DC_DC_DC_DC = 3,
MCPM_CFG_W_CPC_LITE_DC_DC_DC_DC_DC_DC_DC_DC = 4,
MCPM_CFG_WCDMA_IDLE_RX_DC_DC_DC_DC_OFF_DC_DC_DC = 5,
MCPM_CFG_WCDMA_VOICE_DC_DC_DC_DC_OFF_DC_DC_DC = 6,
MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS21_HS11_5M = 7,
MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS42_HS11_10M = 8,
MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS42_HS22_10M = 9,
MCPM_CFG_WCDMA_DATA_DC_DC_DC_DC_OFF_HS63_HS22_10M = 10,
```

```
MCPM_CFG_GSM_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC = 11,
MCPM_CFG_GSM_VOICE_DC_DC_DC_DC_DC_DC_DC_NA = 12,
MCPM_CFG_GSM_DATA_DC_DC_DC_DC_DC_DC_DC_NA = 13,
MCPM_CFG_GSM1_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC = 14,
MCPM_CFG_GSM1_VOICE_DC_DC_DC_DC_DC_DC_DC_NA = 15,
MCPM_CFG_GSM1_DATA_DC_DC_DC_DC_DC_DC_DC_NA = 16,
MCPM_CFG_C_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_NA = 17,
MCPM_CFG_C_STANDBY_DC_DC_DC_DC_DC_DC_DC_NA = 18,
MCPM_CFG_C_VOICE_DC_DC_DC_DC_DC_DC_DC_NA = 19,
MCPM_CFG_C_DATA_DC_DC_DC_DC_DC_DC_DC_NA = 20,
MCPM_CFG_DO_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC = 21,
MCPM_CFG_DO_DATA_DC_DC_DC_DC_DC_DO4_9_DO1_8_DC = 22,
MCPM_CFG_LTE_INIT_DC_DC_DC_DC_DC_DC_DC_DC = 23,
MCPM_CFG_LTE_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_10M = 24,
MCPM_CFG_LTE_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_20M = 25,
MCPM_CFG_LTE_ACQ_DC_DC_DC_DC_DC_DC_DC_DC = 26,
MCPM_CFG_LTE_FDD_DATA_DC_DC_DC_DC_DC_L73_L23_10M = 27,
MCPM_CFG_LTE_TDD_DATA_DC_DC_DC_DC_DC_L73_L23_10M = 28,
MCPM_CFG_LTE_VoLTE_DATA_DC_DC_DC_DC_DC_L73_L23_10M = 29,
MCPM_CFG_LTE_FDD_DATA_DC_DC_DC_DC_DC_L150_L50_20M = 30,
MCPM_CFG_LTE_VoLTE_DATA_DC_DC_DC_DC_DC_L150_L50_20M = 31,
MCPM_CFG_LTE_TDD_DATA_DC_DC_DC_DC_DC_L150_L50_20M = 32,
MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L73_L23_5M_5M = 33,
MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L146_L23_10M_10M = 34,
MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L146_L23_20M_20M = 35,
MCPM_CFG_LTE_DATA_DC_DC_DC_DC_DC_L300_L100_20M_20M = 36,
MCPM_CFG_TDS_STANDBY_DC_DC_DC_DC_DC_DC_DC_DC = 37,
MCPM_CFG_TDS_ACQ_DC_DC_DC_DC_DC_DC_DC_DC = 38,
MCPM_CFG_TDS_IDLE_RX_DC_DC_DC_DC_DC_DC_DC_DC = 39,
MCPM_CFG_TDS_VOICE_DC_DC_DC_DC_DC_DC_DC_DC = 40,
MCPM_CFG_TDS_DATA_DC_DC_DC_DC_DC_DC_DC_DC = 41,
MCPM_CFG_GPS_ACQ_DC_DC_DC_DC_DC_DC_DC_DC = 42,
MCPM_CFG_GPS_NON_DPO_DC_DC_DC_DC_DC_DC_DC_DC = 43,
MCPM_CFG_GPS_DPO_ACTIVE_DC_DC_DC_DC_DC_DC_DC_DC = 44,
MCPM_CFG_A2_ON_DC_DC_DC_DC_DC_DC_DC_DC = 45,
MCPM_CFG_RF_ON_DC_DC_DC_DC_DC_DC_DC_DC = 46,
MCPM_CFG_GSM_CIPHER_ON_DC_DC_DC_DC_DC_DC_DC_DC = 47,
MCPM_CFG_GSM1_CIPHER_ON_DC_DC_DC_DC_DC_DC_DC_DC = 48,
MCPM_MAX_SYS_CFG = 49,
MCPM_NUM_SYS_CFG = 49,
MCPM_SYS_CFG_FORCE32BITS = 2147483647)
```

4 Debug tools – Scripts and flag set

4.1 Modem clock dump

Run the following script in the RPM window:

- MSM8974 – \Modem_proc\mpower\mcpm\scripts\mcpm_clk_dump.cmm
- MDM9x35 – \Modem_proc\mpower\tools\debug_scripts\mcpm_reg_dump.cmm

4.2 How to disable Hexagon power collapse

On MSM8960, Hexagon PC can be disabled by using NV 67202. This is not applicable on later chipsets, such as MSM8974.

```
/* Legacy MCPM NV item, size is fixed to 16 bytes*/
typedef struct {

  uint8  MCPM_NV_QDSS_ENABLE;
  uint8  MCPM_NV_GPIO_Profile_Port;
  uint8  MCPM_NV_UnitTest_Ctrl;
  uint8  MCPM_NV_DISABLE_Q6_PC;
  uint32 MCPM_NV_UnitTest_Commands;
  uint32 MCPM_NV_UnitTest_reserved;
  uint32 MCPM_NV_Q6_Speed_Ctrl;
} MCPM_NV_ITEM_T;
```

Another option is to set the flag sleep_allow_low_power_modes to FALSE.

```
volatile boolean sleep_allow_low_power_modes = TRUE;
```

A References

A.1 Acronyms and terms

Acronym or term	Definition
MCPM	Modem Clock and Power Manager
PL	Product Line