Qualcomm Technologies, Inc.

Qualcomm® Snapdragon™ 820E Processor (APQ8096SGE)

Clock Plan

LM80-P2751-27 Rev. A

February 19, 2018

For additional information or to submit technical questions, go to: https://www.96boards.org/product/dragonboard820c

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

Revision history

Revision	Date	Description
A	February 2018	Initial release

Contents

Figures

Tables

1 Introduction

This document contains a description of the APQ8096SGE ASIC chipset capabilities. Not all features are available, nor are all features supported in the software.

NOTE: Enabling some features may require additional licensing fees.

1.1 Purpose

This document describes the clock plan for the APQ8096SGE ASIC. The document describes the crystals, PLL plans, partition of clock ownership to different processors, and performance levels of network on chips (NoCs), processors, and DDR.

This document is written for APQ8096SGE customers to help them understand how clock plans are implemented in the released software.

1.2 Conventions

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

1.3 Technical assistance

Forum support is available at http://www.96boards.org/forums/forum/products/dragonboard/

2 Clock plan description

This chapter provides details of the clock plan for the APQ8096SGE. Figure 2-1 provides an overview of cores and buses in the chipset. DDR, GMEM, and OCIMEM are memory blocks.

The cores in the chipset are:

- Application processor subsystem (APSS)
- Low-power audio subsystem (LPASS)
- Sensor low-power island (SLPI)
- Multimedia subsystem (MMSS)
- Resource power manager (RPM)

The NoC represents interconnects that connect subsystems and blocks.

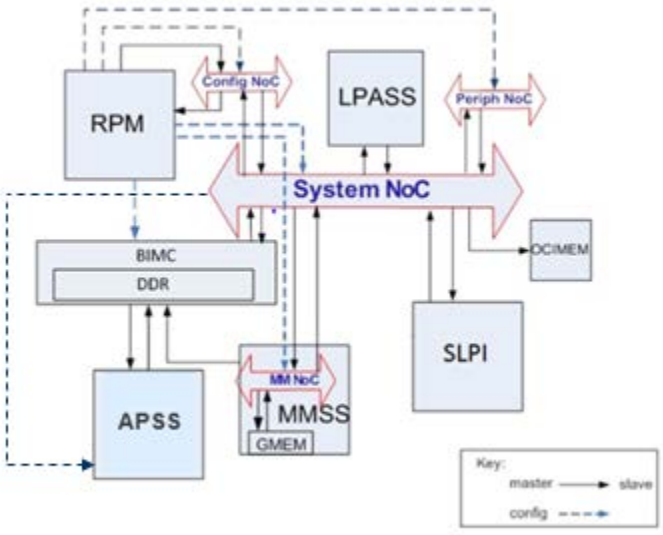

Figure 2-1 APQ8096SGE bus and core topology

2.1 Source crystals

The APQ8096SGE chip has one crystal oscillator (XO) with a frequency of 19.2 MHz. It is the source for all PLLs and is also used as the source for other clocks.

The APQ8096SGE does not use a separate sleep crystal. The sleep clock (with an oscillation frequency of 32.7645 kHz) that is used as the source for the timetick is generated from XO using a divisor of 586.

2.2 Clock ownership

This section lists the clocks by ownership. Only the subsystem that is assigned ownership of a clock can control the clock.

2.2.1 RPM-owned clocks

The RPM owns and manages globally shared resource clocks, such as NoCs and DDR. The following figure shows the RPM-owned clocks and their connectivity to System NoC. Other cores use messages to make requests to enable, disable, or set a minimum rate for the clocks.

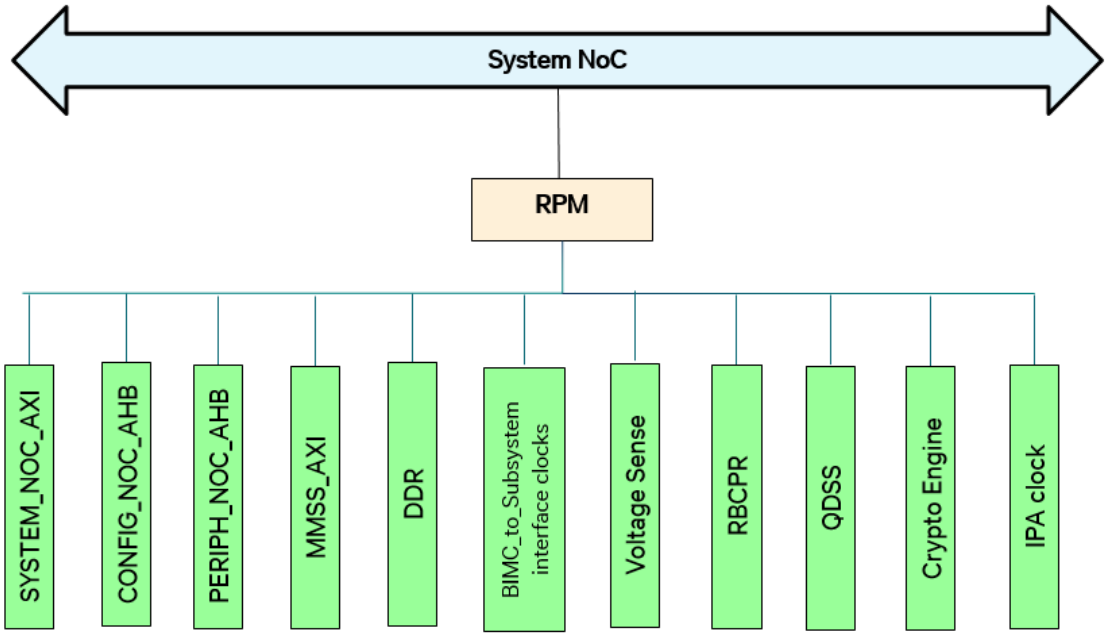

Figure 2-2 RPM-owned clocks

For RPM clock subsystems descriptions, see Appendix B.

2.2.2 APSS-owned clocks

The APSS owns and manages the clock modules displayed in Figure 2-3. The following figure shows the APSS-owned clocks and their connectivity to System NoC through AXI Bus.

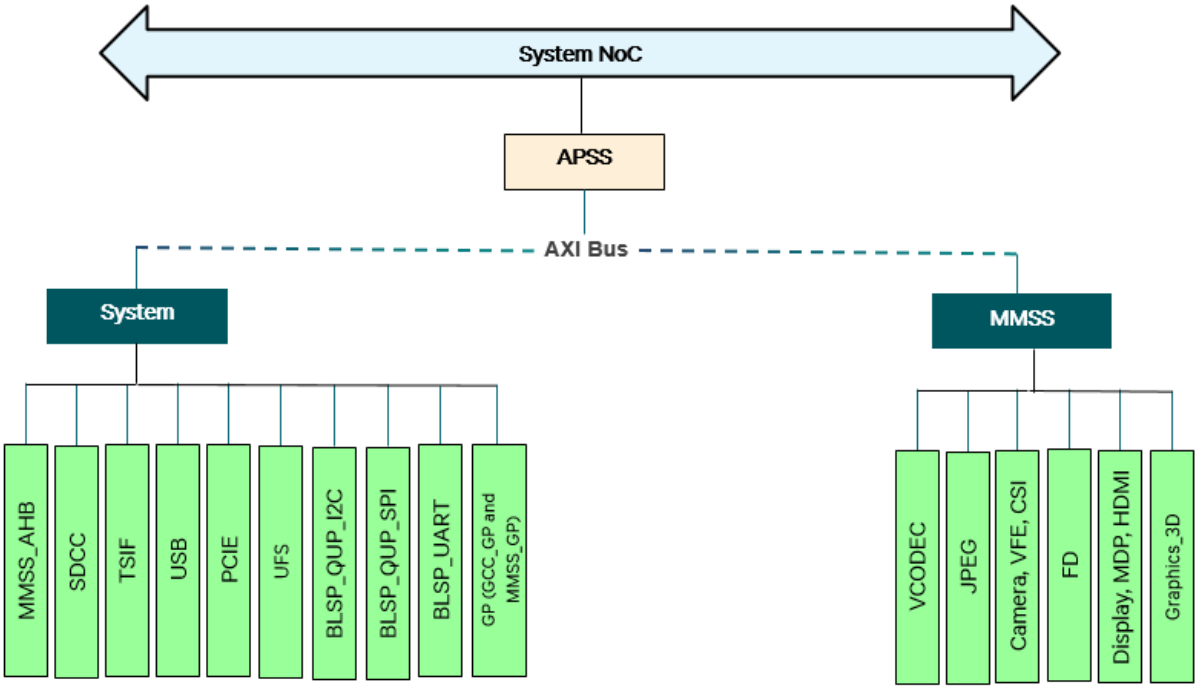

Figure 2-3 APSS-owned clocks

For clock subsystems descriptions of System related and MMSS-related clocks, see Appendix B.

NOTE: BLSP and GCC_GP clock ownership can be partitioned and moved to other processors, depending on the configuration.

2.2.3 LPASS-owned clocks

The following figure shows the LPASS-owned clocks and their connectivity to System NoC:

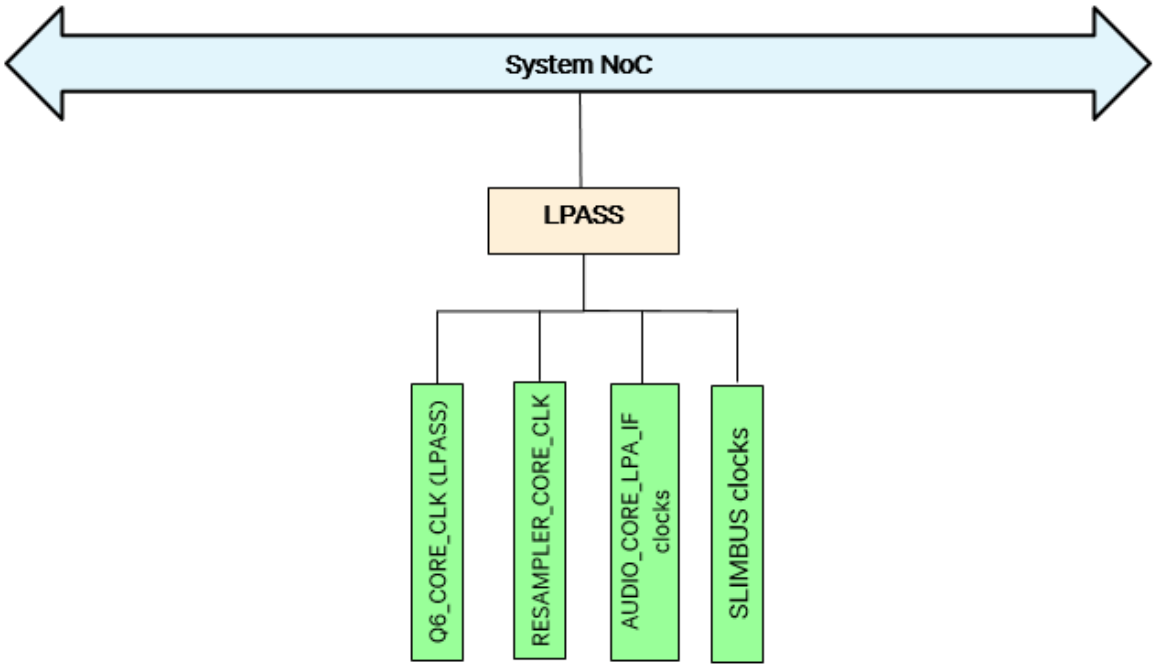

Figure 2-4 LPASS-owned clocks

For clock subsystems descriptions, see Appendix B.

2.3 PLL configuration

The main PLLs, which are partitioned to subsystems, are listed in Table 2-1. The core that owns each PLL configures it and uses it.

The clocks to the different subsystems are derived from these PLLs depending upon the frequency requested by the subsystem. The PLLs are configured during boot time and the clock frequencies cannot be changed.

Table 2-1 PLLs for APQ8096SGE

PLL	Source	Output frequency (MHz)	Notes	Owner Subsystem
GPLL0	XO	600.00	General purpose	RPM
GPLL1	XO	533.00	BIMC	RPM
GPLL2	XO	Varies	BIMC/DDR	RPM
GPLL3	XO	Varies	BIMC/DDR	RPM
GPLL4	XO	384.00	eMMC	APSS
MMPLL0	XO	800.00	MMSS general purpose	RPM

PLL	Source	Output frequency (MHz)	Notes	Owner Subsystem
MMPLL1	XO	810.00	MMSS NoC	APSS
MMPLL2	XO	Varies	GPU	APSS
MMPLL3	XO	1066.00	MMSS general purpose	APSS
MMPLL4	XO	960.00	MMSS general purpose	APSS
MMPLL5	XO	825.00	MDP	APSS
MMPLL8	XO	Varies	GPU	APSS
MMPLL9	XO	1248.00	GPU	APSS
LPAPLL0	XO	614.40	LPASS general purpose	LPASS
LPAPLL1	XO	Varies	LPASS core	LPASS
LPAPLL2	XO	614.40	LPASS general purpose	LPASS
APC0PLL	XO	Varies	Qualcomm® Kryo™ CPU Silver cluster CPU core/L2	APSS
APC1PLL	XO	Varies	Gold cluster CPU core/L2	APSS
CBFPLL	XO	Varies	Coherent bus fabric (CBF)	APSS
SSCPLL0	XO	250.0	Sensor	SLPI
SSCPLL1	XO	Varies	Sensor	SLPI

The RPM owns the bus and DDR clocks and selects a performance level that meets all outstanding requests for a bus. Requests may come from any client.

2.3.1 RPM-managed resources

Performance level refers to the clocks and their supporting frequency levels.

2.3.1.1 System NoC performance levels

The Table 2-2 provides the details related to the various system NoC performance levels, frequencies, and respective source crystals.

Table 2-2 System NoC performance levels

Performance level	Frequency (MHz)	Source
0	9.60	XO
1	19.20	XO
2	50.00	GPLL0
3	100.00	GPLL0
4	150.00	GPLL0
5	200.00	GPLL0
6	240.00	GPLL0

2.3.1.2 Configuration NoC performance levels

The Table 2-3 provides the details related to the various configuration NoC performance levels, frequencies, and respective source crystals.

Table 2-3 Configuration NoC perofrmance levels

Performance level	Frequency (MHz)	Source
0	19.20	XO
1	37.50	GPLL0
2	75.00	GPLL0

2.3.1.3 Peripheral NoC performance levels

The Table 2-4 provides the details related to the various peripheral NoC performance levels, frequencies, and respective source crystals.

Table 2-4 Peripheral NoC performance levels

Performance level	Frequency (MHz)	Source
0	19.20	XO
1	37.50	GPLL0
2	50.00	GPLL0
3	75.00	GPLL0
4	100.00	GPLL0

2.3.1.4 MM NoC performance levels

The Table 2-5 provides the details related to the various MM NoC performance levels, frequencies, and respective source crystals.

Table 2-5 MM NoC performance levels

Performance level	Frequency (MHz)	Source
0	19.20	XO
1	75.00	GPLL0
2	171.43	GPLL0
3	320.00	MMPLL0
4	400.00	MMPLL0

2.3.1.5 Memory (DDR) performance levels

Each core uses DDR with bus interface memory controller (BIMC) supported interfaces.

The Table 2-6 provides the details related to the various DDR performance levels, frequencies, and respective source crystals.

Table 2-6 DDR performance levels

Performance level	Frequency (MHz)	Source
0	100.00	GPLL0
1	150.00	GPLL0
2	200.00	GPLL2/GPLL3
3	300.00	GPLL0
4	412.80	GPLL2/GPLL3
5	547.20	GPLL2/GPLL3
6	681.60	GPLL2/GPLL3
7	768.00	GPLL2/GPLL3
8	1017.60	GPLL2/GPLL3
9	1296.00	GPLL2/GPLL3
10	1555.20	GPLL2/GPLL3
11	1804.80	GPLL2/GPLL3

2.3.2 APSS-managed resources

2.3.2.1 MM NoC performance levels

The Table 2-7 provides the details related to the various MM NoC performance levels, frequencies, and respective source crystals.

Table 2-7 MM NoC performance levels for APSS-managed resources

Performance level	Frequency (MHz)	Source
0	19.20	XO
1	40.00	GPLL0
2	80.00	MMPLL0

2.4 Processor performance levels

Performance level refers to the clocks and their supporting frequency levels.

2.4.1 RPM processor performance levels

The Table 2-8 provides details related to the various RPM processor (ARM Cortex-M3-based processor) performance levels and their frequencies.

Table 2-8 RPM processor performance levels

Performance level	Frequency (MHz)	Source
0	19.20	XO
1	300.00	GPLL0
2	400.00	GPLL0
3	600.00	GPLL0

2.4.2 APSS performance levels

The APSS performance levels provide clock plan data for the cores (Silver and Gold clusters), their L2 cache, and CBF.

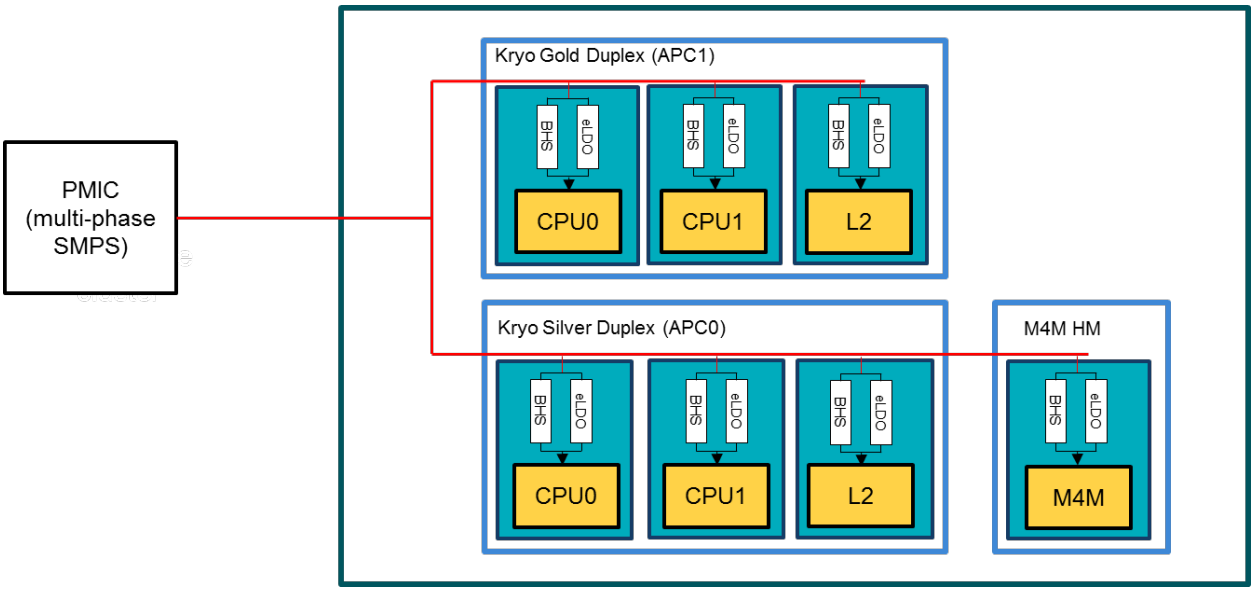

Figure 2-5 APSS clock plan data for Silver and Gold clusters

2.4.2.1 Kryo silver cluster CPU core performance levels

The Table 2-9 provides the processor cores and L2 cache performance levels for the Silver cluster and their frequencies for APSS core 0 and memory.

Silver cluster cores are the power-saving cores.

Table 2-9 Kryo Silver cluster CPU core performance levels

Performance level	Frequency (MHz)	Source
0	307.20	APC0PLL
1	384.00	APC0PLL
2	460.80	APC0PLL
3	537.60	APC0PLL
4	614.40	APC0PLL
5	691.20	APC0PLL
6	768.00	APC0PLL
7	844.80	APC0PLL
8	902.40	APC0PLL
9	979.20	APC0PLL
10	1056.00	APC0PLL
11	1132.80	APC0PLL
12	1209.60	APC0PLL
13	1286.40	APC0PLL
14	1363.20	APC0PLL
15	1440.00	APC0PLL
16	1516.80	APC0PLL
17	1593.60	APC0PLL
18	1785.60	APC0PLL
19	2188.80	APC0PLL

2.4.2.2 Kryo gold cluster CPU core performance levels

The Table 2-10 provides the processor cores and L2 cache performance levels for the Gold cluster and their frequencies for the APSS core 1 and memory.

Gold cluster cores are the performance saving cores.

Table 2-10 Kryo Gold cluster CPU core performance levels

Performance level	Frequency (MHz)	Source
0	307.20	APC1PLL
1	384.00	APC1PLL
2	460.80	APC1PLL
3	537.60	APC1PLL
4	614.40	APC1PLL
5	691.20	APC1PLL

Performance level	Frequency (MHz)	Source
6	748.80	APC1PLL
7	825.60	APC1PLL
8	902.40	APC1PLL
9	979.20	APC1PLL
10	1056.00	APC1PLL
11	1132.80	APC1PLL
12	1209.60	APC1PLL
13	1286.40	APC1PLL
14	1363.20	APC1PLL
15	1440.00	APC1PLL
16	1516.80	APC1PLL
17	1593.60	APC1PLL
18	1670.40	APC1PLL
19	1747.20	APC1PLL
20	1824.00	APC1PLL
21	1900.80	APC1PLL
22	1977.60	APC1PLL
23	2054.40	APC1PLL
24	2150.40	APC1PLL
25	2246.40	APC1PLL
26	2342.40	APC1PLL

2.4.2.3 CBF performance levels

The Table 2-11 provides the CBF performance levels and their frequencies for APSS core 0 and memory. This interconnect keeps data used by the Gold and Silver clusters in sync.

Table 2-11 CBF performance levels

Performance level	Frequency (MHz)	Source
0	192.00	CBFPLL
1	307.20	CBFPLL
2	384.00	CBFPLL
3	441.60	CBFPLL
4	537.60	CBFPLL
5	614.40	CBFPLL
6	691.20	CBFPLL
7	768.00	CBFPLL
8	844.80	CBFPLL
9	902.40	CBFPLL
10	979.20	CBFPLL
11	1056.00	CBFPLL
12	1132.80	CBFPLL

Performance level	Frequency (MHz)	Source
13	1190.40	CBFPLL
14	1286.40	CBFPLL
15	1363.20	CBFPLL
16	1440.00	CBFPLL
17	1516.80	CBFPLL
18	1593.60	CBFPLL

2.4.3 LPASS performance levels

The Table 2-12 provides LPASS performance levels and their frequencies

Table 2-12 LPASS performance levels

Performance level	Frequency (MHz)	Source
0	19.20	XO
1	124.80	LPAPLL1
2	297.60	LPAPLL1
3	480.00	LPAPLL1
4	652.80	LPAPLL1
5	825.60	LPAPLL1

2.4.4 SLPI performance levels

The Table 2-13 provides the SLPI performance levels and their frequencies

Table 2-13 SLPI performance levels

Performance level	Frequency (MHz)	Source
0	19.2	XO
1	125.0	SSCPLL0
2	250.0	SSCPLL0
3	550.0	SSCPLL1
4	700.0	SSCPLL1

2.5 Multimedia module performance levels

The Figure 2-2 shows the various modules connected to MMSS NoC, which include the following.

- GPU
- Video core
- MDSS

Graphics performance levels and clock are controlled by the RPM. The rest of the resources are controlled by the APSS.

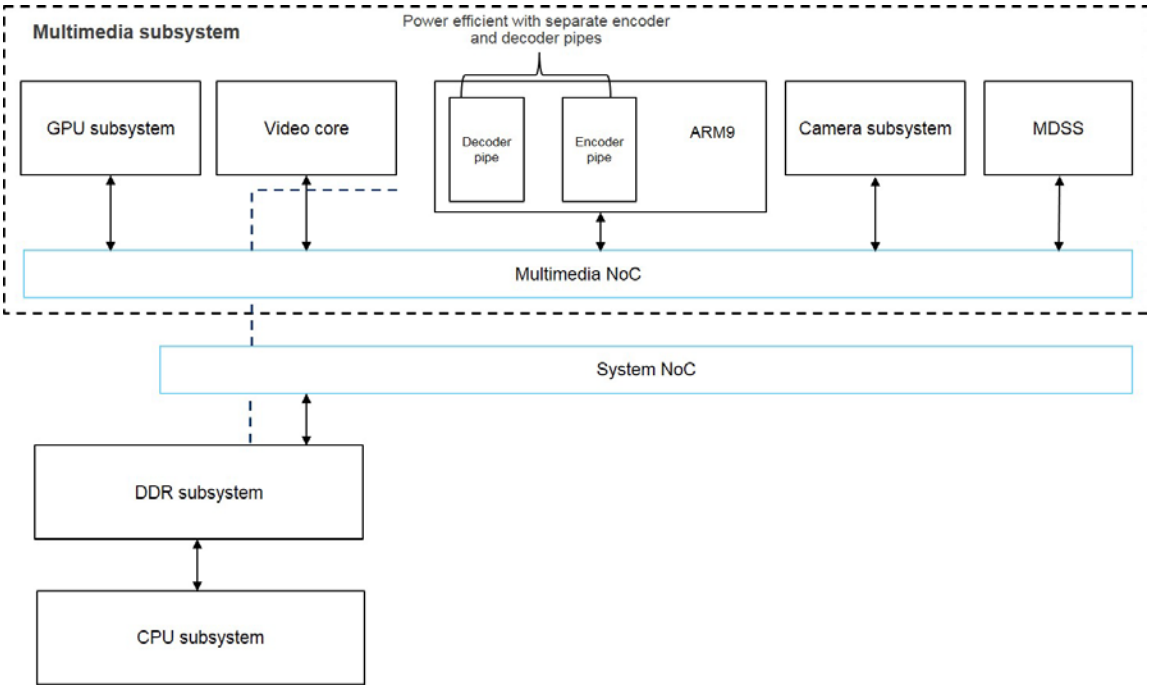

Figure 2-6 Multimedia subsystem modules

2.5.1 GPU performance levels

The Table 2-14 provides the GPU performance levels with their frequencies.

Table 2-14 GPU performance levels

Performance level	Frequency (MHz)	Source
0	133.00	MMPLL2/MMPLL8
1	214.00	MMPLL2/MMPLL8
2	315.00	MMPLL2/MMPLL8
3	401.80	MMPLL2/MMPLL8
4	510.00	MMPLL2/MMPLL8
5	560.00	MMPLL2/MMPLL8
6	624.00	MMPLL9
7	652.80	MMPLL9

2.5.2 Video core performance levels

The Table 2-15 provides the performance levels for the video core with their frequencies

Table 2-15 Video core performance levels

Performance level	Frequency (MHz)	Source
0	75.00	GPLL0
1	150.00	GPLL0
2	346.66	MMPLL3
3	520.00	MMPLL3

2.5.3 MDSS performance levels

The Table 2-16 provides the MDM sub system performance levels and their frequencies

Table 2-16 MDSS performance levels

Performance level	Frequency (MHz)	Source
0	85.71	GPLL0
1	100.00	GPLL0
2	150.00	GPLL0
3	171.43	GPLL0
4	200.00	GPLL0
5	275.00	MMPLL5
6	300.00	GPLL0
7	330.00	MMPLL5
8	412.50	MMPLL5

3 Clock APIs

Refer to the following .c and .h files for the APIs mentioned in this section.

```
/kernel/drivers/clk/clk.c
/kernel/drivers/clk/msm/clock-gcc-8996.c
/kernel/drivers/clk/msm/clock-mmss-8996.c
/kernel/arch/mips/include/asm/clock.h
```

NOTE: Errors are defined in the following header file:

```
/kernel/include/uapi/asm-generic/errno-base.h
```

3.1 clk_get ()

Use this API to return a structure clk corresponding to the clock producer.

```
struct clk *clk_get(struct device *dev, const char *id);
```

Return value: valid structure; or valid IS_ERR() condition containing the error.

3.2 clk_prepare ()

Use this API to prepare a clock source.

`clk_prepare` can go to sleep, which differentiates it from `clk_enable`.

`clk_prepare` can be used instead of `clk_enable` to ungate a clk if the operation goes to sleep.

`clk_prepare` must be called before `clk_enable`.

```
int clk_prepare(struct clk *clk);
```

Return value: 0 on success; or ERROR otherwise.

3.3 clk_unprepare ()

Use this API to undo preparation of a clock source

`clk_unprepare` can go to sleep, which differentiates it from `clk_disable` API.

`clk_unprepare` can be used instead of `clk_disable` to gate a clk if the operation goes to sleep,

`clk_disable` must be called before `clk_unprepare`.

```
void clk_unprepare(struct clk *clk);
```

3.4 clk_enable ()

Use this API to enable a clock.

`Clk_enable` cannot sleep, which differentiates it from `clk_prepare` API.

`clk_enable` can be used instead of `clk_prepare` to ungate a clock.

In a complex case, a clk ungated operation requires both fast and slow part.

Therefore, `clk_enable` and `clk_prepare` are not mutually exclusive.

`clk_prepare` must be called before `clk_enable`.

```
Int clk_enable(struct clk *clk);
```

Return value: 0 on success; or ERROR otherwise.

3.5 clk_disable ()

Use this API to disable a clock.

`clk_disable` cannot sleep, which differentiates it from the `clk_unprepare` API.

`clk_disable` can be used instead of `clk_unprepare` to gate a clock.

In a complex case, a clk gate operation requires a fast and slow part.

Therefore, `clk_unprepare` and `clk_disable` are not mutually exclusive.

`clk_disable` must be called before `clk_unprepare`.

```
void clk_disable(struct clk *clk);
```

3.6 clk_prepare_enable ()

Use this API to prepare a clock and then enabled it in a single sequence non-atomic context.

```
static inline int clk_prepare_enable(struct clk *clk);
```

3.7 clk_disable_unprepare ()

Use this API to disable and unprepare the specified clock in a single sequence non-atomic context.

```
static inline void clk_disable_unprepare(struct clk *clk);
```

3.8 clk_set_rate ()

Use this API to specify a new rate for clock.

```
int clk_set_rate(struct clk *clk, unsigned long rate);
```

Return value: 0 on success or ERROR otherwise

3.9 clk_get_rate ()

Use this API to verify or get the clock rate.

```
unsigned long clk_get_rate(struct clk *clk);
```

Return value: returns the rate of the clock.

Example code

```
struct clk *gp1_clk;
gp1_clk = clk_get(&cust_gcc_dev, "gcc_gp1_clk");
if (IS_ERR(gp1_clk))
{
  printk("%s: Get gp1_clk error!!!\n", __func__);
  gp1_clk = NULL;
  return;
}
pr_err("%s: clk_set_rate \n", __func__);
clk_set_rate(gp1_clk, 100000000);

pr_err("%s: clk_prepare_enable \n", __func__);
clk_prepare_enable(gp1_clk);
```

4 Clock debugging using ADB commands

Debug commands in this section are applicable for Android.

To view a list of all the clocks on the device:

```
root@msm8996:cd /sys/kernel/debug/clk/
root@msm8996:ls
```

4.1 Start ADB interface

Use the following commands to enable, disable, and measure the different subsystem clocks.

```
adb devices
adb root
adb wait-for-devices
adb remount
adb shell
```

4.2 Debug GCC GP1 clock

Use the following command to debug `gcc_gp1_clk`.

```
cd /sys/kernel/debug/clk/gcc_gp1_clk
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat list_rates
      19200000
      100000000
      200000000
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat print
cxo_clk_src
gp1_clk_src
            CMD_RCGR: 0x80000000
            CFG_RCGR: 0x00000000
               M_VAL: 0x00000000
               N_VAL: 0x00000000
               D_VAL: 0x00000000
gcc_gp1_clk
                CBCR: 0x80000000
```

```
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat measure
0
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat enable
0
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # echo 1 > enable
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat enable
1
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat measure
19200109
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat print
cxo_clk_src
gp1_clk_src
            CMD_RCGR: 0x00000000
            CFG_RCGR: 0x00000000
               M_VAL: 0x00000000
               N_VAL: 0x00000000
               D_VAL: 0x00000000
gcc_gp1_clk
                CBCR: 0x00000001
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # echo 0 > enable
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat enable
0
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat measure
0
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk # cat print
cxo_clk_src
gp1_clk_src
            CMD_RCGR: 0x80000000
            CFG_RCGR: 0x00000000
               M_VAL: 0x00000000
               N_VAL: 0x00000000
               D_VAL: 0x00000000
gcc_gp1_clk
                CBCR: 0x80000000
root@msm8996:/sys/kernel/debug/clk/gcc_gp1_clk #
```

4.3 Debug 3D graphics clock

Use the following command to debug the 3D graphics clock `gfx3d_clk_src_v2`.

```
cd /sys/kernel/debug/clk/gfx3d_clk_src_v2
root@msm8996:/sys/kernel/debug/clk/ gfx3d_clk_src_v2 # cat print
cxo_clk_src
mmsscc_xo
mmpll2
            PLL_MODE: 0x00000000
           PLL_L_VAL: 0x00000014
       PLL_ALPHA_VAL: 0x55555600
     PLL_ALPHA_VAL_U: 0x000000d5
        PLL_USER_CTL: 0x01200301
      PLL_CONFIG_CTL: 0x4001051b
mmpll2_out_main
mmpll2_postdiv_clk
gfx3d_clk_src_v2
            CMD_RCGR: 0x80000000
            CFG_RCGR: 0x00000300
root@msm8996:/sys/kernel/debug/clk/gfx3d_clk_src_v2 # cat enable
0
root@msm8996:/sys/kernel/debug/clk/gfx3d_clk_src_v2 # echo 1 > enable
root@msm8996:/sys/kernel/debug/clk/gfx3d_clk_src_v2 # cat enable
1
root@msm8996:/sys/kernel/debug/clk/gfx3d_clk_src_v2 # cat rate
133000000
root@msm8996:/sys/kernel/debug/clk/gfx3d_clk_src_v2 # echo 56000000 >
rate
root@msm8996:/sys/kernel/debug/clk/gfx3d_clk_src_v2 # cat rate
560000000
root@msm8996:/sys/kernel/debug/clk/gfx3d_clk_src_v2 # cat print
cxo_clk_src
mmsscc_xo
mmpll8
            PLL_MODE: 0x80000007
           PLL_L_VAL: 0x0000001d
       PLL_ALPHA_VAL: 0xaaaaab00
```

```
     PLL_ALPHA_VAL_U: 0x0000002a
        PLL_USER_CTL: 0x01200001
      PLL_CONFIG_CTL: 0x4001051b
mmpll8_out_main
mmpll8_postdiv_clk
gfx3d_clk_src_v2
            CMD_RCGR: 0x00000002
            CFG_RCGR: 0x00000400
root@msm8996:/sys/kernel/debug/clk/gfx3d_clk_src_v2 #
```

4.4 Debug system NoC clock

Use the following command to debug the system NoC `clock snoc_clk.`

```
cd /sys/kernel/debug/clk/snoc_clk
root@msm8996:/sys/kernel/debug/clk/snoc_clk # ls
root@msm8996:/sys/kernel/debug/clk/snoc_clk # cat measure
100000897
root@msm8996:/sys/kernel/debug/clk/snoc_clk # cat enable
1
root@msm8996:/sys/kernel/debug/clk/snoc_clk # echo 240000000 > rate
root@msm8996:/sys/kernel/debug/clk/snoc_clk # cat measure
240001940
root@msm8996:/sys/kernel/debug/clk/snoc_clk # cat print
snoc_clk
root@msm8996:/sys/kernel/debug/clk/snoc_clk #
```

NOTE: `snoc_clk` is 100 MHz because the peripherals (USB debugging cable connected) have requested the same clock so that the RPM aggregates all the clocks requested by peripherals and sets the maximum of that.

A References

For additional details about APQ8096SGE, click:
https://www.96boards.org/product/dragonboard820c/

B Acronyms and terms

Term	Definition
AHB	Advanced high-performance bus
APSS	Application processor subsystem
AXI	Advanced eXtensible Interface
BAM	Bus access manager
BIMC	Bus interface memory controller
BLSP	BAM low-speed peripheral
CBF	Coherent bus fabric
CE	Crypto engine
CPP	Camera postprocessor
CSI	Camera serial interface
DDR	Double data rate (memory)
GPU	Graphics processing unit
IPA	Internet packet accelerator
LPASS	Low-power audio subsystem
MDSS	Multimedia display subsystem
MMSS	Multimedia subsystem
NoC	Network on chip
PLL	Phase-lock loop
QDSS	Qualcomm debug subsystem
RPM	Resource power manager
SDCC	Secure digital card controller
SLPI	Sensor low-power island
SPMI	System power management interface
SVS	Static voltage scaling
TSIF	Transport stream interface
VDD_APC	Power rail voltage for APSS
VDD_CX	Core power rail voltage
VDD_EBI	Power rail voltage for DDR
VDD_GFX	Power rail voltage for graphics (GFX)
VDD_MX	Power rail voltage for memory
VFE	Video front end
XO	Crystal oscillator

C Subsystem descriptions

Subsystem	Description
AUDIO_CORE_LPA_IF clocks	Audio subsytem configuration
BIMC_to_Subsystem interface clocks	Bus-integrated memory controller subsystem configuration
BLSP_QUP_I2C	BAM low-speed peripheral and Qualcomm Universal Peripheral Serial interintergrated circuit
BLSP_QUP_SPI	BAM low-speed peripheral and Qualcomm Universal Peripheral serial peripheral interface
BLSP_UART	BAM low-speed peripheral universal asynchronous receiver/transmitter
CE	Crypto engine subsystem configuration
CONFIG_NOC_AHB	Advanced high performance bus subsystem configuration
DDR	DDR (memory) subsystem configuration
GP	General-purpose controller for global clock and MM
IPA ()	Internet packet accelerator clock subsystem configuration
MMSS_AHB	Multimedia subsystem advanced high performance bus configuration
MMSS_AXI	Multimedia subsystem configurations through Advanced eXtensible Interface
PCIE	Peripheral component interconnect express subsystem configuration
PERIPH_NOC_AHB	Advanced high performance bus configuration for peripheral NoC
Q6_CORE_CLK	LPASS subsystem configuration
QDSS	Qualcomm debug subsystem configuration
RBCPR	Rapid bridge core power reduction subsystem configuration
RESAMPLER_CORE_CLK	Video processing subsystem configuration
SDCC	Secure digital card controller subsystem configuration
SLIMBUS clocks	Serial low-power interchip media bus subsystem configuration
SYSTEM_NOC_AXI	System NoC configuration through Advanced eXtensible Interface
TSIF	Transport stream interface subsystem configuration
UFS	UFS subsystem configuration
USB	USB subsystem configuration
VSense	Voltage sense configurations

EXHIBIT 1

PLEASE READ THIS LICENSE AGREEMENT ("AGREEMENT") CAREFULLY. THIS AGREEMENT IS A BINDING LEGAL AGREEMENT ENTERED INTO BY AND BETWEEN YOU (OR IF YOU ARE ENTERING INTO THIS AGREEMENT ON BEHALF OF AN ENTITY, THEN THE ENTITY THAT YOU REPRESENT) AND QUALCOMM TECHNOLOGIES, INC. ("QTI" "WE" "OUR" OR "US"). THIS IS THE AGREEMENT THAT APPLIES TO YOUR USE OF THE DESIGNATED AND/OR ATTACHED DOCUMENTATION AND ANY UPDATES OR IMPROVEMENTS THEREOF (COLLECTIVELY, "MATERIALS"). BY USING OR COMPLETING THE INSTALLATION OF THE MATERIALS, YOU ARE ACCEPTING THIS AGREEMENT AND YOU AGREE TO BE BOUND BY ITS TERMS AND CONDITIONS. IF YOU DO NOT AGREE TO THESE TERMS, QTI IS UNWILLING TO AND DOES NOT LICENSE THE MATERIALS TO YOU. IF YOU DO NOT AGREE TO THESE TERMS YOU MUST DISCONTINUE AND YOU MAY NOT USE THE MATERIALS OR RETAIN ANY COPIES OF THE MATERIALS. ANY USE OR POSSESSION OF THE MATERIALS BY YOU IS SUBJECT TO THE TERMS AND CONDITIONS SET FORTH IN THIS AGREEMENT.

1.1 **License.** Subject to the terms and conditions of this Agreement, including, without limitation, the restrictions, conditions, limitations and exclusions set forth in this Agreement, Qualcomm Technologies, Inc. ("QTI") hereby grants to you a nonexclusive, limited license under QTI's copyrights to use the attached Materials; and to reproduce and redistribute a reasonable number of copies of the Materials. You may not use Qualcomm Technologies or its affiliates or subsidiaries name, logo or trademarks; and copyright, trademark, patent and any other notices that appear on the Materials may not be removed or obscured. QTI shall be free to use suggestions, feedback or other information received from You, without obligation of any kind to You. QTI may immediately terminate this Agreement upon your breach. Upon termination of this Agreement, Sections 1.2-4 shall survive.

1.2 **Indemnification.** You agree to indemnify and hold harmless QTI and its officers, directors, employees and successors and assigns against any and all third party claims, demands, causes of action, losses, liabilities, damages, costs and expenses, incurred by QTI (including but not limited to costs of defense, investigation and reasonable attorney's fees) arising out of, resulting from or related to: (i) any breach of this Agreement by You; and (ii) your acts, omissions, products and services. If requested by QTI, You agree to defend QTI in connection with any third party claims, demands, or causes of action resulting from, arising out of or in connection with any of the foregoing.

1.3 **Ownership.** QTI (or its licensors) shall retain title and all ownership rights in and to the Materials and all copies thereof, and nothing herein shall be deemed to grant any right to You under any of QTI's or its affiliates' patents. You shall not subject the Materials to any third party license terms (e.g., open source license terms). You shall not use the Materials for the purpose of identifying or providing evidence to support any potential patent infringement claim against QTI, its affiliates, or any of QTI's or QTI's affiliates' suppliers and/or direct or indirect customers. QTI hereby reserves all rights not expressly granted herein.

1.4 **WARRANTY DISCLAIMER.** YOU EXPRESSLY ACKNOWLEDGE AND AGREE THAT THE USE OF THE MATERIALS IS AT YOUR SOLE RISK. THE MATERIALS AND TECHNICAL SUPPORT, IF ANY, ARE PROVIDED "AS IS" AND WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESS OR IMPLIED. QTI ITS LICENSORS AND AFFILIATES MAKE NO WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THE MATERIALS OR ANY OTHER INFORMATION OR DOCUMENTATION PROVIDED UNDER THIS AGREEMENT, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OR AGAINST INFRINGEMENT, OR ANY EXPRESS OR IMPLIED WARRANTY ARISING OUT OF TRADE USAGE OR OUT OF A COURSE OF DEALING OR COURSE OF PERFORMANCE. NOTHING CONTAINED IN THIS AGREEMENT SHALL BE CONSTRUED AS (I) A WARRANTY OR REPRESENTATION BY QTI, ITS LICENSORS OR AFFILIATES AS TO THE VALIDITY OR SCOPE OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT OR (II) A WARRANTY OR REPRESENTATION BY QTI THAT ANY MANUFACTURE OR USE WILL BE FREE FROM INFRINGEMENT OF PATENTS, COPYRIGHTS OR OTHER INTELLECTUAL PROPERTY RIGHTS OF OTHERS, AND IT SHALL BE THE SOLE RESPONSIBILITY OF YOU TO MAKE SUCH DETERMINATION AS IS NECESSARY WITH RESPECT TO THE ACQUISITION OF LICENSES UNDER PATENTS AND OTHER INTELLECTUAL PROPERTY OF THIRD PARTIES.

1.5 **LIMITATION OF LIABILITY.** IN NO EVENT SHALL QTI, QTI'S AFFILIATES OR ITS LICENSORS BE LIABLE TO YOU FOR ANY INCIDENTAL, CONSEQUENTIAL OR SPECIAL DAMAGES, INCLUDING BUT NOT LIMITED TO ANY LOST PROFITS, LOST SAVINGS, OR OTHER INCIDENTAL DAMAGES, ARISING OUT OF THE USE OR INABILITY TO USE, OR THE DELIVERY OR FAILURE TO DELIVER, ANY OF THE MATERIALS, OR ANY BREACH OF ANY OBLIGATION UNDER THIS AGREEMENT, EVEN IF QTI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE FOREGOING LIMITATION OF LIABILITY SHALL REMAIN IN FULL FORCE AND EFFECT REGARDLESS OF WHETHER YOUR REMEDIES HEREUNDER ARE DETERMINED TO HAVE FAILED OF THEIR ESSENTIAL PURPOSE. THE ENTIRE LIABILITY OF QTI, QTI's AFFILIATES AND ITS LICENSORS, AND THE SOLE AND EXCLUSIVE REMEDY OF YOU, FOR ANY CLAIM OR CAUSE OF ACTION ARISING HEREUNDER (WHETHER IN CONTRACT, TORT, OR OTHERWISE) SHALL NOT EXCEED US$10.

2. **COMPLIANCE WITH LAWS; APPLICABLE LAW.** You agree to comply with all applicable local, international and national laws and regulations and with U.S. Export Administration Regulations, as they apply to the subject matter of this Agreement. This Agreement is governed by the laws of the State of California, excluding California's choice of law rules.

3. **CONTRACTING PARTIES.** If the Materials are downloaded on any computer owned by a corporation or other legal entity, then this Agreement is formed by and between QTI and such entity. The individual accepting the terms of this Agreement represents and warrants to QTI that they have the authority to bind such entity to the terms and conditions of this Agreement.

4. **MISCELLANEOUS PROVISIONS.** This Agreement, together with all exhibits attached hereto, which are incorporated herein by this reference, constitutes the entire agreement between QTI and You and supersedes all prior negotiations, representations and agreements between the parties with respect to the subject matter hereof. No addition or modification of this Agreement shall be effective unless made in writing and signed by the respective representatives of QTI and You. The restrictions, limitations, exclusions and conditions set forth in this Agreement shall apply even if QTI or any of its affiliates becomes aware of or fails to act in a manner to address any violation or failure to comply therewith. You hereby acknowledge and agree that the restrictions, limitations, conditions and exclusions imposed in this Agreement on the rights granted in this Agreement are not a derogation of the benefits of such rights. You further acknowledges that, in the absence of such restrictions, limitations, conditions and exclusions, QTI would not have entered into this Agreement with You. Each party shall be responsible for and shall bear its own expenses in connection with this Agreement. If any of the provisions of this Agreement are determined to be invalid, illegal, or otherwise unenforceable, the remaining provisions shall remain in full force and effect. This Agreement is entered into solely in the English language, and if for any reason any other language version is prepared by any party, it shall be solely for convenience and the English version shall govern and control all aspects. If You are located in the province of Quebec, Canada, the following applies: The Parties hereby confirm they have requested this Agreement and all related documents be prepared in English.