Application Note

AN001

MINIMIZING JITTER ON CLOCKS GENERATED BY THE GAP8 APPLICATION PROCESSOR

Version: Rel.1.0

Date: March, 2020

Author: M Guermandi

GREENWAVES TECHNOLOGIES

MINIMIZING JITTER ON GENERATED CLOCKS IN GAP8 APPLICATION PROCESSOR

This application note is meant to provide a basic characterization of the cycle-to-cycle jitter for periodic signals generated at the output of GAP8 peripherals. Data is measured and presented for the clock signal generated at the output of the I2S peripheral I2S0 at two reference frequencies of 1 MHz and 4 MHz. The application note also introduces strategies to minimize jitter.

1. ORIGIN OF JITTER IN GAP8

Two main mechanisms contribute to the jitter of signals generated at the output of GAP8 peripherals. The first one is the jitter of the integrated ultra-low-power crystal oscillator and sets the floor of the achievable jitter. The second mechanism is due to noise being injected on the VDD pin and being converted on jitter at the output of the FLL.

2. JITTER MEASUREMENTS IN DIFFERENT CONFIGURATIONS

2.1. Jitter obtained with clean power supply to the SOC (VDD)

The following tables present the jitter measured when a clean DC voltage is provided as supply to the SOC for, respectively, 1 MHz and 4 MHz output clock signal frequency. Jitter is not significantly affected by the DC value of the supply voltage while it varies depending on the FLL frequency.

Those results may also be used as reference for jitter levels when the internal DC-DC is used as power supply to the SOC, provided further down this document.

Table 1: Cycle to cycle jitter with clean VDD and 1 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
10	1430
20	1220
50	1080
100	730
150	560
200	540
250	450

Table 2: Cycle to cycle jitter with clean VDD and 4 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
50	410
100	283
150	223
200	211
250	207

To improve jitter at lower FLL frequencies, the FLL can be made run at higher frequency and its output divided (see the Appendix of this document for details). This will lead to an increase in power consumption in the FLL but not on the core.

Following tables present the measured jitter adopting this strategy, with the values of the FLL frequencies and division factors for the different SOC operating frequencies.

Table 3: Cycle to cycle jitter with clean VDD and 1 MHz output signal

SOC FREQUENCY [MHz]	FLL FREQUENCY [MHz]	FLL DIVISION FACTOR	CYCLE-TO-CYCLE JITTER [ppm]
10	80	8	1010
20	160	8	760
50	200	4	600
100	200	2	580
150	150	1	560
200	200	1	540
250	250	1	450

In general, the FLL is more sensitive to noise generated at or below the crystal oscillator frequency. High frequency noise in the range of MHz such as that typical of DC/DC converters operating in PWM mode is well tolerated while significant ripples in the KHz range might provide a steep increase in the measured jitter.

2.2. Jitter obtained with internal DC/DC power supply to the SOC, UNFILTERED :

When providing power supply from the internal DC/DC converter, jitter can increase at low loads due to the DC/DC switching to noisier PFM mode. It is not recommended to generate high frequency signals requiring low jitter when in PFM mode, unless proper filtering is provided at the output of the DC/DC (see following section). In these conditions, jitter is not a random noise with Gaussian distribution and zero mean. This can lead to noise being accumulated from cycle to cycle, leading to instability in the frequency of the generated signals.

The switching from PFM mode to PWM mode occurs at 17 mA output current. However, an hysteresis is provided so that once the DC/DC is in PWM mode it will switch back to PFM mode at 10 mA load. When the DC/DC is in PWM mode, jitter values are in line with those presented in the previous section (up to 10% higher).

Following tables present jitter **when DC/DC is in PFM mode** (noisiest mode) for 1 V, 1.1 V, 1.2 V SOC supply voltage, except at the highest frequencies for 1.2 V supply voltage (200 and 250 MHz) where required power draw already forces the DC/DC to operate in **PWM mode** (as flagged in the tables)

Table 4: Cycle to cycle jitter with VDD = 1V from DC/DC in PFM mode and 1 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
10	16000
20	15700
50	14000
100	10800
150	8290

Table 5: Cycle to cycle jitter with VDD = 1V from DC/DC in PFM mode and 4 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
50	14100

100	11400
150	8400

Table 6: Cycle to cycle jitter with VDD = 1.1 V from DC/DC in PFM mode and 1 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
10	11500
20	11800
50	10440
100	7630
150	5900
200	4880

Table 7: Cycle to cycle jitter with VDD = 1.1 V from DC/DC in PFM mode and 4 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
50	10600
100	7960
150	6240
200	5160

Table 8: Cycle to cycle jitter with VDD = 1.2 V from DC/DC in PFM mode and 1 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
10	11700
20	12220
50	10600
100	7490
150	5840
200	5050
250	510 (PWM MODE)

Table 9: Cycle to cycle jitter with VDD = 1.2 V from DC/DC in PFM mode and 4 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
50	7800
100	5640
150	4160
200	960 (PWM MODE)
250	840 (PWM MODE)

The presented values are obtained mostly when the DC/DC is in PFM mode, which is better in terms of conversion efficiency at low current but presents significant switching noise at or below the reference clock frequency, making the FLL operation noisier. **Operating the DC-DC in PWM mode provides significantly better clock jitter results.**

Note: there is an hysteresis mechanism on the switching between PFM and PWM mode. Starting from PFM mode, the DC-DC will switch to PWM mode when the load exceeds 17mA. However, once in PWM, it will stay in that mode unless the current load drops down to below 10mA.

This means that if the signal is generated after a computational intensive task, jitter won't increase if the load current stays above 10 mA. Alternatively, if the code allows it, the DC/DC can be forced to operate in PWM mode with a "dummy" high load task which is executed right before signal generation (e.g. a simple cluster mount and unmount or increasing the SOC frequency temporarily).

Also, starting with Rev.C of GAP8, a bit will allow to force the internal DC-DC in PWM mode. This is intended to obtain better jitter results at light load (low current) at the expense of lower conversion efficiency in that region.

2.3. Jitter with internal DC/DC power supply to the SOC , FILTERED

For most applications requiring a low jitter but operating at low loads (below 10 mA), a simple RC filtering at the input of SOC power supply is enough to reduce the jitter to acceptable levels.

A typical filter that can be adopted is a compact, first order RC filter with R = 0.3 Ohm and C = 100 uF. Higher values for R might reduce jitter further, however the loss on the resistor should be taken into account when evaluating power efficiency of the solution and the voltage drop associated to higher load task evaluated so as not to reduce effective supply voltage below the minimum required values for the operating frequency.

Following tables again present some jitter results **when DC/DC is in PFM mode** (noisiest mode) for 1 V, 1.1 V, 1.2 V SOC supply voltage, this time with VDD filtering at DC-DC output – except, again, at the highest frequencies for 1.2 V supply voltage (200 and 250 MHz) where required power draw already forces the DC/DC to operate in **PWM mode** (as flagged in the tables).

Table 10: Cycle to cycle jitter with VDD = 1V from filtered DC/DC in PFM mode and 1 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
10	3140
20	3040
50	2620
100	1920
150	1500

Table 11: Cycle to cycle jitter with VDD = 1V from filtered DC/DC in PFM mode and 4 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
50	3080
100	2210
150	1720

Table 12: Cycle to cycle jitter with VDD = 1.1 V from filtered DC/DC in PFM mode and 1 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
10	2450
20	2590
50	2200
100	1570
150	1130
200	971

Table 13: Cycle to cycle jitter with VDD = 1.1 V from filtered DC/DC in PFM mode and 4 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
50	2600
100	1840
150	1400
200	1180

Table 14: Cycle to cycle jitter with VDD = 1.2 V from filtered DC/DC in PFM mode and 1 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
10	3060
20	3120
50	2860
100	2070
150	1550
200	1420
250	690 (PWM MODE)

Table 15: Cycle to cycle jitter with VDD = 1.2 V from filtered DC/DC in PFM mode and 4 MHz output signal

SOC/FLL FREQUENCY [MHz]	CYCLE-TO-CYCLE JITTER [ppm]
50	2600
100	1800
150	1480
200	1200
250	700 (PWM MODE)

Best results are still obtained in PWM mode rather than PFM mode, however the filtering improves clock jitter at light loads (DC-DC in PFM mode).

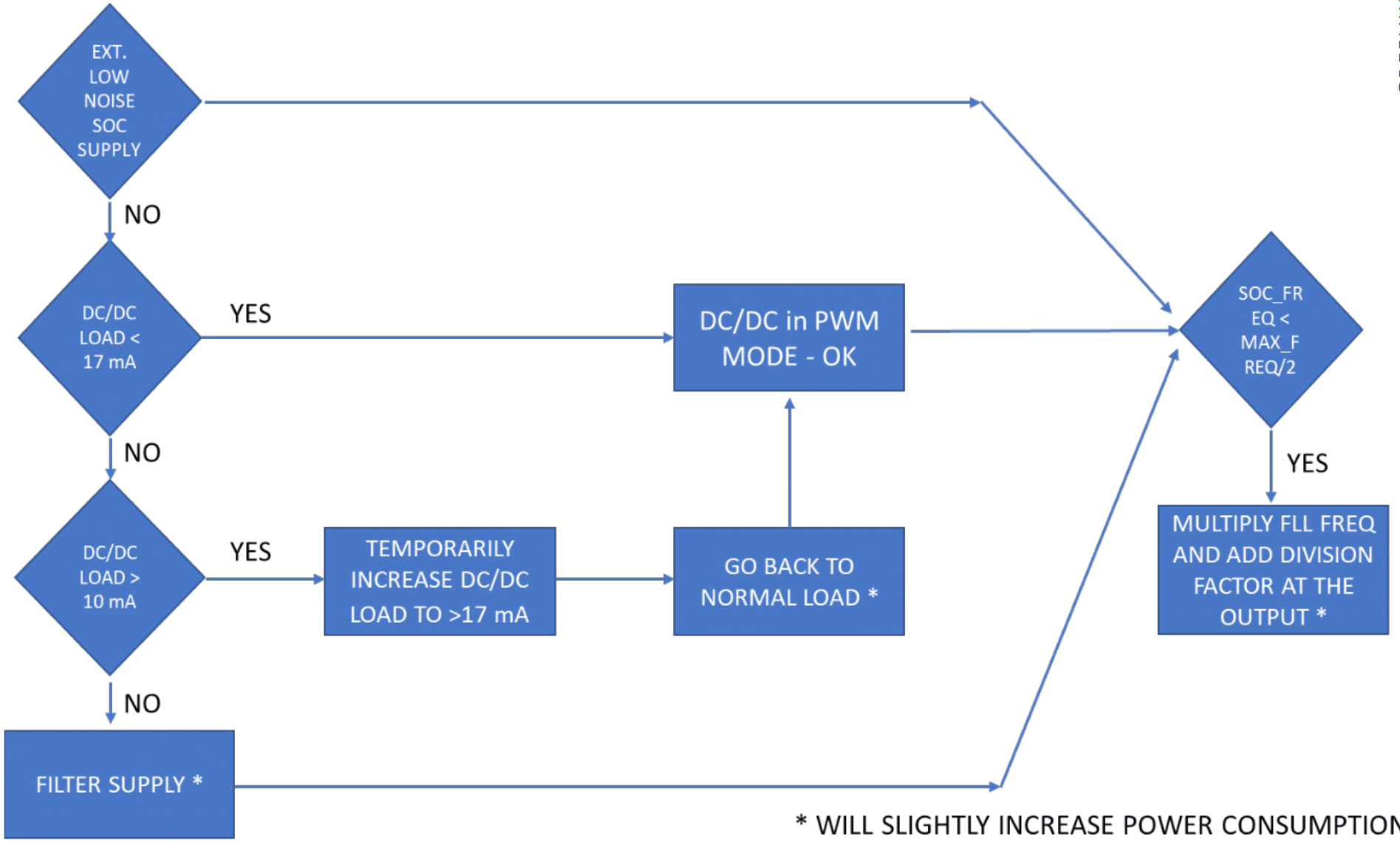
As introduced in section 2.1, to improve jitter at lower FLL frequencies, the FLL can be made run at higher frequency and its output divided (see the Appendix of this document for details). This will lead to an increase in power consumption in the FLL but not on the core.

3. CONCLUSION

The internal DC-DC of GAP8 introduces some jitter on generated clocks used by peripherals, for example I2S. This jitter is more significant at low current load because the DC-DC then operates in a mode that generates a more noisy power supply.

Whether the jitter present on those clocks is acceptable or not depends on application constraints. To help select the best system configuration, this AN provides measurements of jitter in different “worst case” configurations, that is when the internal DC-DC operates at low current load and therefore is in PFM mode.

The following flowchart shows how to optimize jitter on GAP8 generated clocks depending on the conditions under which the application runs.



* WILL SLIGHTLY INCREASE POWER CONSUMPTION

APPENDIX

The GAP8 register controlling the division factor at the output of the FLL is the **SOC FLL CFG1** register at address **0x1A100004**, bits 29:26.

GAP8 SDK functions for configuring FLL will set the field to its default value of 1 and set the frequencies accordingly. The values to be set in the bit field to further divide the FLL output frequency are:

DIVISION FACTOR	0x1A100004, bits 29:26
1	0001
2	0010
4	0011
8	0100

For example, the following code snippet will set GAP8 to operate at 50 MHz by setting the FLL output at 200 MHz and dividing the output frequency by 4:

```
rt_freq_set(RT_FREQ_DOMAIN_FC, 200000000);  //SET FLL TO RUN AT 200 MHz

pulp_write32(0x1A100004,((*(uint32_t*)0x1A100004)&0xC3FFFFFF)+0x0C000000);  //DIVIDE BY 4
```

Please note that the SDK functions used subsequently will assume that GAP8 is operating at the FLL output frequency (200 MHz in the example).