Frequency Margining with FemtoClock 2

FemtoClock 2 is a high-performance clock generator with I2C access. I2C can be used to modify any setting, including settings that affect the frequency of the output clocks.

When FemtoClock 2 starts up, it first moves its configuration from OTP (ROM) memory to operation registers (RAM). Circuits in FemtoClock 2 look at the registers for their settings. I2C has access to the registers so I2C can be used to modify the behavior from the original configuration that was loaded from OTP memory. When powering down and then powering back up, the original configuration in OTP memory will be restored into the registers.

When changing frequencies, however, the user will need to modify the frequency divider values. Sometimes the user will also need to modify other parameters in addition to the divider values. For example, when the APLL frequency changes more than a certain amount, the APLL must be recalibrated after the change.

Contents

1. FemtoClock 2 Block Diagram

The following figure shows a typical block diagram as displayed in a FemtoClock 2 datasheet.

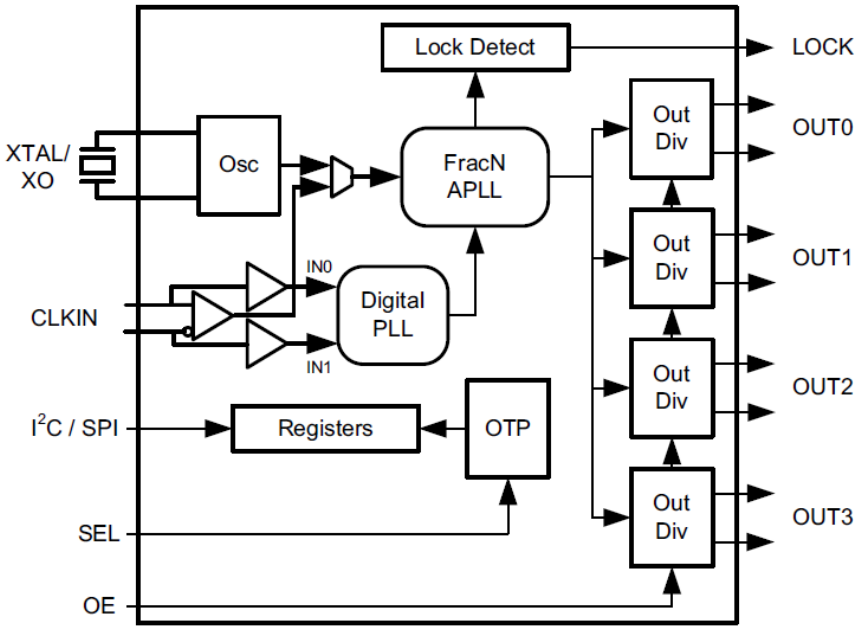

Figure 1. FemtoClock 2 Block Diagram

The reference frequency from a crystal or CLKIN is first multiplied up in the FracN APLL, and is controlled by the FracN Feedback Divider. The high APLL frequency of around 10GHz is then divided down to the outputs with integer output dividers.

OUTn = XTAL × 2 × FB-Div / OutDiv-n

There are certain limitations:

- The APLL has a limited frequency range of 9.7GHz to 10.7GHz. The APLL frequency is XTAL × 2 × FB-Div. The frequency of an output OUTn will be APLL / OutDiv-n.
- The output dividers can only divide with integer numbers so the frequency changes with relatively large steps when changing an output divider. The APLL feedback divider is a fractional-N type with a 27-bit fraction and can control frequencies with better than 0.1ppb resolution. When changing the feedback divider, all output frequencies will change. When changing an output divider, only the frequency of one output will change.
- When changing the APLL feedback divider, the APLL frequency changes and a re-calibration of the APLL may be needed. When the FemtoClock 2 device first starts up, it will calibrate its APLL. When modifying the APLL frequency from the start-up frequency with more than 0.1% or 1,000ppm, a re-calibration is required. Re-calibrating is not glitch-free. Outputs will be disabled for the duration of the re-calibration. Small frequency changes, inside a ±1,000ppm window, do not require re-calibration and will be glitch-free. Changing output dividers is also glitch-free and does not require re-calibration.

2. I2C Addressing

FemtoClock 2 has more than 256 registers so the usual 1 byte for the register address in I2C is not sufficient. To access all registers there are two different methods:

1. “Paging” with 256 registers in Page 0 and remaining registers in Page 1.
2. “2 Byte Addressing” where 2 bytes are used for the register address.

All registers required for frequency changes are in Page 1. For the examples in this document, the Paging method will be used. The FemtoClock 2 device will start up with Page 0 selected. For the examples, we first change the Page to 1 and then write to the registers for changing the frequencies.

For changing to Page 1, write: `FC 00 01 10 20`

This is a “block write” of multiple registers. 0xFC is the start address. 00# is written to 0xFC, 01# to 0xFD, 10# to 0xFE and 20# to 0xFF. The 01# in 0xFD is the actual page. The rest are conditions that need to be written together with the page number.

For changing back to Page 0, write: `FC 00 00 10 20`

These four registers, 0xFC~FF, are transparent between the pages and can be accessed on every page.

3. Registers

The following is a list of important registers for changing frequencies:

- Output 0, Integer Divider – 14 bits total, Bits[7:0] in register 0x100[7:0] and Bits[13:8] in register 0x101[5:0]
- Output 1, Integer Divider – 14 bits total, Bits[7:0] in register 0x108[7:0] and Bits[13:8] in register 0x109[5:0]
- Output 2, Integer Divider – 14 bits total, Bits[7:0] in register 0x110[7:0] and Bits[13:8] in register 0x111[5:0]
- Output 3, Integer Divider – 14 bits total, Bits[7:0] in register 0x118[7:0] and Bits[13:8] in register 0x119[5:0]

Please note that Bit[7] in registers 0x101, 0x109, 0x111, and 0x119 enables the LDO for the specific output divider. When it is needed to access bits in the range 13:8, make sure to also write Bit[7]=1 to keep the LDO enabled. This will only be required for output divider values above 255 or output frequencies roughly below 40MHz.

Feedback Divider Fraction: 27 bits total, Bits[7:0] in register 0x150[7:0]
Bits[15:8] in register 0x151[7:0]
Bits[23:16] in register 0x152[7:0]
Bits[26:24] in register 0x153[2:0]

Feedback Divider Integer: 10 bits total, Bits[7:0] in register 0x154[7:0] and Bits[9:8] in register 0x155[1:0].

The unused bits in registers 0x153 and 0x155 can be left at 0.

The Feedback Divider values are atomic. This means that the whole value (all registers) needs to be written for the change to take effect. When updating the fraction, please write all four registers (0x150, 0x151, 0x152, and 0x153) to update the fraction value, even when for example, register 0x153 will not actually change.

When the frequency of the APLL VCO changes to a value outside of ±0.1% (roughly ±10MHz) from the previously calibrated frequency, please trigger a new APLL calibration.

Re-calibrate the APLL VCO: Bit[7] in register 0x162. Toggle to 1 to trigger the APLL VCO calibration. Please follow up with a toggle back to 0 to set it up for the next calibration. With most configurations, the remainder of register 0x162 is all zeros so an APLL VCO re-calibration requires writing 62 80, followed by writing 62 00 (assuming we are already in Page 1).

The above re-calibration is just the pure VCO calibration. There is also a trigger for a more thorough re-initialization of the APLL, mentioned in the datasheet as “apll_reinit”. When toggling this bit, in addition to re-calibrating the APLL VCO, the dividers are also reset to realign phase relationships.

APLL Re-initialization: Bit[0] in register 0x00A. Make sure to read back the content of register 0x00A before attempting the re-initialization, to make sure to only toggle bit[0]. Please also note that this bit is in Page 0.

3.1 Changing an Output Frequency with the Output Divider

The easiest method for changing an output frequency is to change the output divider for that output. Each output can be controlled individually because each output has its own output divider. Frequency changes are glitch-less and easy. For output frequencies above about 40MHz, only the lower 8 bits of the output divider are used and only one register needs to be rewritten. For example, the RC22504A works best with a 78.125MHz crystal for 156.25MHz or 312.5MHz outputs, where the VCO is 10GHz. The output dividers will be 64 for 156.25MHz and 32 for 312.5MHz. Assuming the doubler between crystal and phase detector is enabled, the phase detector frequency will be 156.25MHz and the feedback divider is 64.

When changing an output divider from 64 to 63, the output frequency increases to 10,000 / 63 = 158.73MHz or +1.6%. This 1.6% is the smallest step size and is the resolution for changing a 156.25MHz output, using the output divider. So the disadvantage of using an output divider to change an output frequency is that the resolution is relatively large and changes with the output frequency. For example, when the output frequency is 312.5MHz with an output divider value of 32, the resolution or step size is 3.1%.

Example 1, assuming the APLL VCO is 10.000GHz

1. Write `FC 00 01 10 20` to change the register address page to 1.
2. To change OUT0 = 156.25MHz to 158.73MHz (+1.6%), write `00 3F` (3F# = 63 decimal).
3. To change OUT1 = 156.25MHz to 158.73MHz (+1.6%), write `08 3F` (3F# = 63 decimal).
4. To change OUT0 = 156.25MHz to 153.85MHz (-1.5%), write `00 41` (41# = 65 decimal).
5. To make OUT2 = 312.5MHz, write 10 20 (20# = 32 decimal).
6. To make OUT2 = 100MHz, write 10 64 (64# = 100 decimal).
7. To make OUT2 = 125MHz, write 10 50 (50# = 80 decimal).
8. To make OUT3 = 1GHz, write 18 0A (0A# = 10 decimal).

As you can see, this method is also useful to quickly change an output frequency to anything that can be divided down from 10GHz with an integer value.

Example 2: Ethernet FEC rates

For Ethernet, sometimes the clock is increased from a standard 156.25MHz or 312.5MHz with a ratio of 33/32 to operate at the FEC rate. 156.25MHz is increased to 161.1328125MHz and 312.5MHz is increased to 322.265625MHz. When setting up the APLL with VCO = 10.3125MHz, it is easy to switch outputs between standard and FEC rate frequencies, as follows:

1. Write `FC 00 01 10 20` to change the register address page to 1.
2. To make OUT0 = 156.25MHz, write 00 42 (42# = 66 decimal).
3. To make OUT1 = 161.1328125MHz, write 08 40 (40# = 64 decimal).
4. To make OUT2 = 312.5MHz, write 10 21 (21# = 33 decimal).
5. To make OUT3 = 322.265625MHz, write 18 20 (20# = 32 decimal).

3.2 Changing Output Frequencies with the APLL Feedback Divider

The APLL Feedback Divider is a Fractional Divider with 27 bits precision for the fraction. This allows for frequency settings with around 0.1ppb or 0.0001ppm resolution.

For the best phase noise performance, Renesas recommends to configure the APLL with an integer value for the feedback divider. When trying to make small, precise frequency changes, it is easier to work with fractions for the APLL feedback divider. For example, the RC22504A works best with a 78.125MHz crystal for 156.25MHz or 312.5MHz outputs, where the VCO is 10GHz. The output dividers will be 64 for 156.25MHz and 32 for 312.5MHz. Assuming the doubler between crystal and phase detector is enabled, the phase detector frequency will be 156.25MHz and the feedback divider is 64.

In order to use the Feedback Divider for relatively small and precise frequency changes, the Feedback Divider must be operated in Fractional mode. Together with changing the feedback divider to fractional mode, a number of other parameters also must be changed to realign the optimization for fractional mode operation.

When the original configuration uses an integer for the feedback divider, and control of the output frequencies with feedback divider fractions is desired, please write the following:

1. First write `FC 00 01 10 20` to change the register address page to 1.
2. Write `56 23 0D 44 3E 64 27` to reconfigure the feedback divider.

0x156 = 23# changes the feedback divider to fractional mode with bits[1:0] and enables Sigma Delta dithering with bit[5].
0x157 = 0D# reconfigures the frequency doubler with bit[3].
0x158 = 44# reconfigures the charge pump current.
0x159 = 3E# reconfigures a charge pump current offset between up and down currents.
0x15A = 64# and 0x15B = 27# reconfigure loop filter resistors and capacitors.

This update should not have changed the output frequency. At this point the APLL and its feedback divider are prepared for controlling fractions of the feedback divider. The feedback divider is now in “Fractional Mode” and other circuits like the charge pump and loop filter are realigned for optimum performance in Fractional Mode.

Now we can rewrite the feedback divider fraction value and perhaps the feedback divider integer value to change the output frequencies.

Example: All outputs are 156.25MHz, the VCO is 10GHz and the crystal is 78.125MHz. The objective is to increase each output with exactly +1% to 157.8125MHz.

The original APLL feedback divider is 10,000 / 78.125 / 2 = 64.
We want to change that to 1.01 x 64 = 64.64 (+1%).
We only need to change the fraction to 0.64 to make all output frequencies go up with 1%.
The fraction register setting must be 0.64 × 2^27 = 85,899,345.92.
The number is then rounded up to 85,899,346 = 05 1E B8 52 hex. The rounding causes a tiny 0.009ppb error.
To show the insignificance of the rounding error, the exact change including the error is: +1.000000009%.

1. Write `FC 00 01 10 20` to change the register address page to 1.
2. Write `56 23 0D 44 3E 64 27` to change the APLL feedback divider operation to Fractional Mode.
3. Write `50 52 B8 1E 05` to change the fraction value to 0.64.
4. Write `62 80` to re-calibrate the APLL VCO.
5. Write `62 00` to reset the calibration bit.

All outputs should now be toggling at 157.8125MHz.

From here to all outputs at -1% or 154.6875MHz requires the feedback divider to be reprogrammed to 0.99 × 64 = 63.36 (-1%). Now we also need to update the integer value of the feedback divider.

The integer register setting will be 63 or 3F#.

The Fraction register setting will be 0.36 x 2^27 = 48,318,382.08. We round that down to 48,318,382. The hex value will be 02 E1 47 AE.

Hint: To convert decimal to hex values, use Microsoft Excel® and type DEC2HEX(48318382,8). The “8” is for 8 hex characters.

The integer registers are right behind the fraction registers so this update can be done with one block write:

1. Write `50 AE 47 E1 02 3F 00` to change the fraction value to 0.36 and the integer value to 63.
2. Write `62 80` to re-calibrate the APLL VCO.
3. Write `62 00` to reset the calibration bit.

All outputs should now be toggling at 154.6875MHz.

3.3 Changing Frequencies with Jitter Attenuator Configuration

The RC32504A and RC32514A can be configured as Jitter Attenuators. What this means is that the DPLL is controlling the APLL to keep the APLL synchronized with the reference clock input. When making a change to the APLL feedback divider, the DPLL must be reprogrammed as well to be synchronizing for the new APLL frequency. For this version of the application note, we will not include the required changes to the DPLL settings when changing the APLL feedback divider. Renesas also recommends to limit output frequency changes with a Jitter Attenuator to changes of the integer output dividers.

- RC22504A and RC22514A Synthesizers → Full control of both APLL feedback divider and output dividers.
- RC32504A and RC32514A Jitter Attenuators → Only control output dividers.
- RC32504A and RC32514A can also be configured as Synthesizers where they operate identically to the RC22504A and RC22514A so you can also control both the APLL feedback divider and output dividers.

4. Revision History

Revision	Date	Description
1.0	Aug 4, 2021	Initial release.

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01 Jan 2024)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.