syn1588 ®

Application Note AN008

Using the “frequency” and “extclk” command and the external jitter-cleaner PLL

Version 1.17.0 - October 2023

Abstract

This application note describes the procedure for using the external jitter-cleaner PLL that can be equipped on a syn1588® PCIe NIC Revision 2.0/1 and syn1588® Dual NIC. This PLL can be used to remove the jitter of digitally generated, synchronized clocks. Furthermore it allows generation of high frequency single-ended clock signals (up to 150 MHz). It can also be used (in case of a syn1588® PCIe NIC Revision 2.1) to generate the NIC internal clock from a user-provided input clock.

The “frequency” command described in this application note can also be used to digitally generated clock signals which is applicable to hardware without an equipped jitter-cleaner PLL. The generated clock signals can be routed to any of the on-board SMA connectors.

The “extclk” command described in this application note is an option for a syn1588® PCIe NIC Revision 2.1 that has to be ordered separately (see AN002 syn1588® PCIe NIC Ordering Options).

Features

- “Frequency” command:
  Generating up to two clock signals
  - Either one signal from the jitter-cleaner PLL and one digitally generated signal
  - Or two digitally generated signals
  - Routing each generated signal to one of the four connectors (X4, X5, X6 or X7).
  - Supported output frequencies (see Table 1 on page 9)
- “Extclk” command:
  Generating the internal clock from a user-provided input signal
  (via connector X8, only applicable for a syn1588® PCIe NIC Revision 2.1)
  - Supported input frequencies (see Table 2 on page 14)

Requirements for the syn1588® PCIe NIC Revision 2.0 or 2.1

- Hardware requirements (these versions have been tested, older versions might still be supported)
  - syn1588® PCIe NIC Revision 2.0/1, equipped with a jitter-cleaner PLL
  - "Frequency" command:
    Firmware 750 (or newer),
    check with, e.g., ptpmmm -> "clock" command (Figure 1, "HW build 757")
  - "Extclk" command:
    Firmware 804 (or newer), applicable only for Revision 2.1
- Software requirements (these versions have been tested, older versions might still be supported)
  - syn1588nic driver 1.4-7
  - syn1588® PTP Stack version 1.2.434 (or newer) with enabled PLL access and management message handling, running on the host equipped with the syn1588® PCIe NIC.
  - ptpmmm (version 1.5 or newer) or syn1588 command line tool (version 1.5 or newer)
  - Contact Oregano Systems support if you do not have the appropriate software version available. Contact Oregano Systems

Requirements for the syn1588® PCIe NIC Revision 2.3

- The version 1.17 of the syn1588® Software Suite will in a first step not support the on-board jitter cleaner PLL.
- This support will be added either with an update of the release 1.18
  - One excerpt from this rule is the ExtClk option using an external 10 MHz clock which is available by default without using the fSync utility; simply connect the appropriate external clock to connector X8.

Requirements for the syn1588® Dual NIC

While the syn1588® Dual NIC is quite similar to two syn1588® PCIe NICs on a single board there are some important differences to be observed. The syn1588® Dual NIC offers two network interfaces which own their own syn1588® hardware clock function. These two syn1588® hardware clock functions have to share the same four user I/Os of the PCIe card. The user is responsible to assign each user I/O just to one function of any of the two syn1588® hardware clocks (associated with the two network interfaces). If one violates this sequence the assignment for interface 2 will supersede. Please check AN036 for details on the syn1588® Dual NIC GPIOs.

Additionally all board related components like the jitter cleaner PLL are just available with interface 1.

Useful documents

- syn1588® User Guide
- syn1588® Clock IP Core Register Map (AN001)
- syn1588® PCIe NIC Quick Start Guide (AN004)
- syn1588® PCIe NIC Firmware Update (AN014)
- syn1588® Dual NIC Special Considerations on GPIOs (AN036)

Overview

The syn1588® PCIe NIC can generate periodic signals and provide them as outputs via any of the SMA connectors of the board. This can be done in two ways:

Digitally generating a periodic signal via the PERIOD 0 and/or 1 register set on the FPGA (syn1588® User Guide, chapter 6.1.1.7 Frequency Output)

Digitally generating a periodic signal in the same way. Feeding it to the external jitter-cleaner PLL. Generating the desired output frequency via this jitter-cleaner PLL.

This feature is further referenced and described in this document as “frequency” command.

The syn1588® PCIe NIC Rev 2.1 can use an external frequency source as input to the jitter-cleaner PLL. One output clock of the jitter-cleaner PLL is connected to the NIC FPGA. The jitter-cleaner PLL can be used to generate the internal 25 MHz clock from this external frequency source (e.g., a 10 MHz output from a rubidium clock).

This feature is further referenced and described in this document as "extclk" command.

Preparation

The jitter-cleaner PLL can be configured via three different methods:

- syn1588
- ptpmmm
- fSync

syn1588

This tool can be used to configure local syn1588® PCIe NICs by issuing the "frequency" command (explained in the chapter "Frequency" command).

ptpmmm

This tool can be used to configure local and remote syn1588® PCIe NICs via PTP management messages. The "frequency" command can be used to configure the external PLL.

Prerequisite for access via ptpmmm: An active syn1588® PTP Stack on all workstations hosting the to-be-configured syn1588® PCIe NICs. This can be verified by using the "clock" command of the ptpmmm command line tool, which will list all connected PTP devices, Figure 1 gives an example.

```
sudo ./ptpmmm -i enp2s0 -v critical
2022-01-07 12:05:34.988340 [INFO    ] [              ]  syn1588(R) PTP Management
Tool - IEEE1588-2008
2022-01-07 12:05:34.988375 [INFO    ] [              ]  Build date: 2022-01-
05T12:43:16 - v1.14-9-g7cbf9e73
2022-01-07 12:05:34.988382 [INFO    ] [              ]  Copyright (c) Oregano
Systems - Design & Consulting GesmbH 2005-2022
2022-01-07 12:05:34.988389 [INFO    ] [              ]  Confidential unpublished
data - All rights reserved
2022-01-07 12:05:34.988396 [INFO    ] [              ]  Important note: This
software cannot manage nor display PTP running on the same node as this PTP
management software.
2022-01-07 12:05:34.988405 [INFO    ] [              ]  Command line:
>clock
2022-01-07 12:06:10.588041 [INFO    ] [              ]  <---- Management
2022-01-07 12:06:10.588776 [INFO    ] [              ]   ----> Management
001EC0FFFE85748B 1 ? "IEEE 802.3" 001EC085748B IPv4:0A00016C FFFFFF "Oregano
Systems; syn1588(R) PCIe NIC; 00:1E:C0:85:74:8B" "HW build 757; syn1588(R) Clock M
2.3.1;  SW 1.2.374" "syn1588_NIC" 001B19000100
0050C2FFFEC2DFAE 1 ? "IEEE 802.3" 0050C2C2DFAE IPv4:0A000102 000F0C "MBG;;" ";;"
";" 001B19000100
001EC0FFFE85942D 1 OC "IEEE 802.3" 001EC085942D IPv4:0A000193 FFFFFF "Oregano
Systems; syn1588(R) PCIe NIC; 00:1E:C0:85:94:2D" "HW build 757; syn1588(R) Clock M
2.3.1;  SW 1.2.434" "syn1588_NIC" 001B19000100
0004A3FFFE4B4101 1 OC "IEEE 802.3" 0004A34B4101 IPv4:0A000111 FFFFFF "Oregano
Systems; syn1588(R) VIP; 00:04:A3:4B:41:01" "HW build 625; syn1588(R) Clock M
2.3.1;  SW 1.2.439" "syn1588_VIP" 001B19000100
0004A3FFFE4A9DD0 1 ? "IEEE 802.3" 0004A34A9DD0 IPv4:0A00017B FFFFFF "Oregano
Systems; syn1588(R) VIP; 00:04:A3:4A:9D:D0" "HW build 623; syn1588(R) Clock M
2.3.1;  SW 1.2.433" "syn1588_VIP" 001B19000100
```

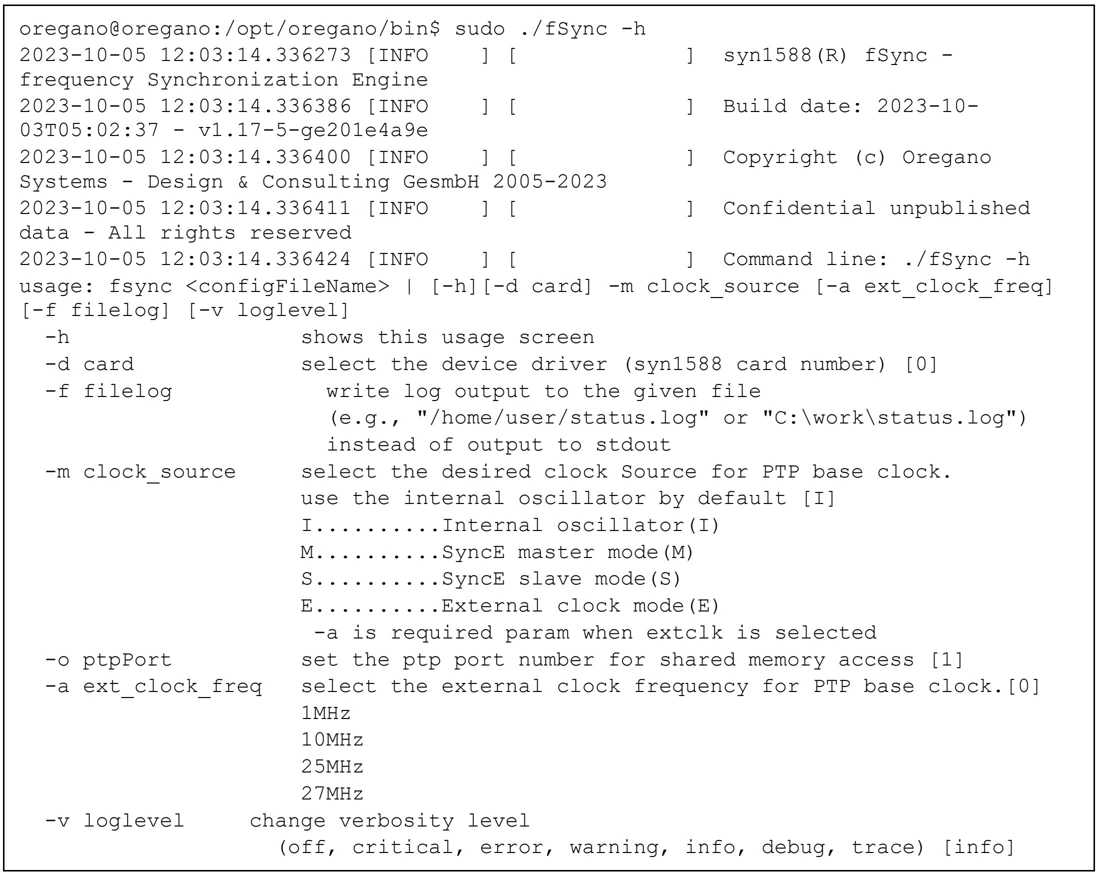
```
>
```

Figure 1: ptpmmm command line tool, clock command

The output of the clock command in Figure 1 shows five PTP devices identified by their clock ID (denoted in bold text) and further information regarding the PTP stack and firmware version (in case of the four listed syn1588® PCIe NIC devices). The last two entries are syn1588® VIP Evaluation Boards, which do not provide an external PLL (HW build < 667) but the generation of a digital frequency is still possible. The first two entries have a firmware version of 757 and a syn1588® PTP Stack version of 1.2.434 (denoted in yellow), which is a valid setup to provide access to the external PLL of these two NICs.

fSync

This is the general purpose frequency synchronization engine to configure local syn1588® PCIe NICs. In the future fSync will allow SyncE clocking options as well. fSync can just be used to configure the “extclk” function. Figure 2 gives an example for the invocation of fSync.



```
oregano@oregano:/opt/oregano/bin$ sudo ./fSync -h
2023-10-05 12:03:14.336273 [INFO    ] [              ]  syn1588(R) fSync -
frequency Synchronization Engine
2023-10-05 12:03:14.336386 [INFO    ] [              ]  Build date: 2023-10-
03T05:02:37 - v1.17-5-ge201e4a9e
2023-10-05 12:03:14.336400 [INFO    ] [              ]  Copyright (c) Oregano
Systems - Design & Consulting GesmbH 2005-2023
2023-10-05 12:03:14.336411 [INFO    ] [              ]  Confidential unpublished
data - All rights reserved
2023-10-05 12:03:14.336424 [INFO    ] [              ]  Command line: ./fSync -h
usage: fsync <configFileName> | [-h][-d card] -m clock_source [-a ext_clock_freq]
[-f filelog] [-v loglevel]
  -h                    shows this usage screen
  -d card               select the device driver (syn1588 card number) [0]
  -f filelog              write log output to the given file
                          (e.g., "/home/user/status.log" or "C:\work\status.log")
                          instead of output to stdout
  -m clock_source       select the desired clock Source for PTP base clock.
                        use the internal oscillator by default [I]
                        I..........Internal oscillator(I)
                        M..........SyncE master mode(M)
                        S..........SyncE slave mode(S)
                        E..........External clock mode(E)
                         -a is required param when extclk is selected
  -o ptpPort            set the ptp port number for shared memory access [1]
  -a ext_clock_freq     select the external clock frequency for PTP base clock.[0]
                        1MHz
                        10MHz
                        25MHz
                        27MHz
  -v loglevel       change verbosity level
                      (off, critical, error, warning, info, debug, trace) [info]
```

Figure 2: fSync command line tool

Please note that the currently supported clock sources are I (internal) and E (extclk).

Hardware Limitations

This chapter gives an overview of the different hardware platforms for which this application note is applicable and about the limitations for these platforms.

syn1588® PCIe NIC Revision 2.0

A syn1588® PCIe NIC Revision 2.0 equipped with a jitter-cleaner PLL (Si5328 as depicted in Figure 3) supports the full functionality of the “frequency” command. The hardware build number depicts, if PERIOD 0 (<= 790) or PERIOD 1 (> 790) is connected to the external PLL. This is important, if the user wants to generate an output signal by the PLL and a second output signal via the FPGA.

If it is not equipped with a jitter-cleaner PLL, the “frequency” command can still be used to digitally generate a frequency via PERIOD 0 and/or 1 and route it to the SMA connectors.

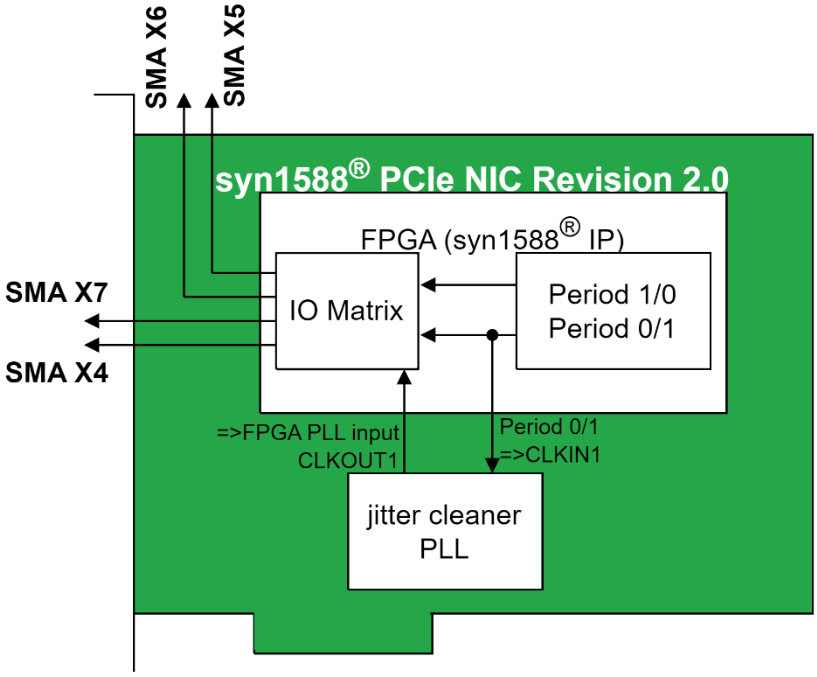


Figure 3: syn1588® PCIe NIC Rev 2.0 SMA connectors

syn1588® PCIe NIC Revision 2.1 & 2.3

A syn1588® PCIe NIC Revision 2.1 supports the full functionality of the “frequency” command. If the NIC has been ordered with the ExtClk option it also supports the “extclk” command.

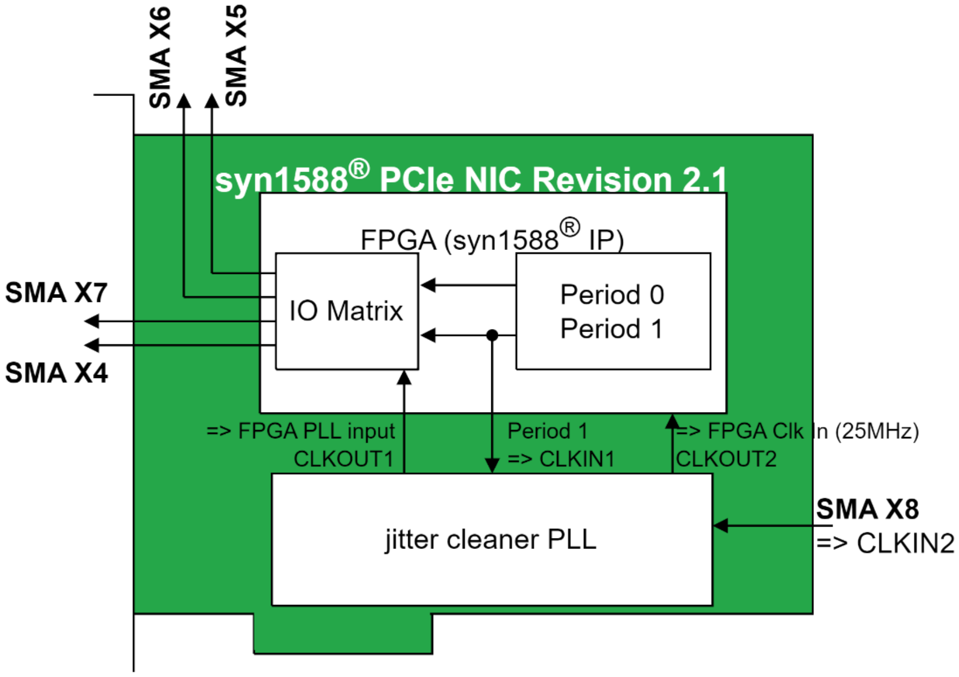


Figure 4: syn1588® PCIe NIC Rev 2.1 & 2.3 SMA connectors

syn1588® Dual NIC

The jitter cleaner PLL on a syn1588® Dual NIC can just be access from the syn1588® hardware clock associated with the first network interface (even CLOCKID or MAC address). All functions using the PLL like the frequency command of the syn1588 utility or the fSync utility can only be invoked on interface 1 of the syn1588® Dual NIC.

Additionally both syn1588® hardware clock functions have to share the same four user I/Os of the PCIe card. The user is responsible to assign each user I/O just to one function of any of the two syn1588® hardware clocks (associated with the two network interfaces). If one violates this sequence the assignment for interface 2 will supersede. Please check AN036 for details on the syn1588® Dual NIC GPIOs.

Frequency output range

Aside from using the jitter-cleaner PLL as frequency signal source, a signal can also be generated digitally.

Frequency	Jitter-Cleaner PLL clock	Digital clock
Minimum	8 kHz	15,25 µHz
Minimum (recommended)	20 kHz	15,25 µHz
Maximum (recommended)	148,5 MHz	<< 15 MHz (firmware 757)
Maximum (differential outputs)	808 MHz	15 MHz (firmware 757)

Table 1:Frequency output range

PLL Recommendations

The jitter-cleaner PLL produces noisy signal edges for low frequencies (i.e., below 20 kHz) output signals. Therefore, it is recommended to generate these and lower frequencies digitally. The typical setup of the syn1588® PCIe NIC includes SMA connectors for single ended signal output. If it is necessary to generate a higher output frequency, contact Oregano regarding a syn1588® PCIe NIC equipped for differential output of the PLL signal.

Digital Clock Recommendations

The maximum frequency is constrained by the current firmware. It is recommended to use a considerably lower frequency than the maximum possible frequency for a digitally generated clock.

“Frequency” command

The “frequency” command is used to generate an output signal with a specified frequency from a specified source on a specified output connector.

The “frequency” command can be used with both the syn1588 and the ptpmmm command line tools. While ptpmmm can be used to remotely configure syn1588® devices, the syn1588 tool only configures the devices of the platform where it is executed.

The syntax is equivalent for both tools with a single exception. For ptpmmm three additional parameters are needed, denoting the target clock device (e.g., the syn1588® PCIe NIC) in the network for the command.

Command syntax ptpmmm:

```
frequency [ptpmmm: clkid] [ptpmmm: port] [ptpmmm: domain]
          [Par0: d|p] [Par1: 4|5|6|7] [Par2: 0|1] [Par3: <frequency>]
          or
          [Par0: f] [Par1: 4|5|6|7] [Par: <filename>]
```

Command syntax syn1588:

```
frequency [Par0: d|p|f] [Par1: 4|5|6|7] [Par2: 0|1] [Par3: <frequency>]
          or
          [Par0: f] [Par1: 4|5|6|7] [Par4: <filename>]
```

Parameters

Par0

This parameter selects between digitally generated (“d”) and external PLL generated clock signal (“p”), or loading a PLL configuration from a file (“f”).

Par1

This parameter selects the output connector to which the generated signal is routed. The different settings are 4 = X4, 5 = X5, 6 = X6 and 7 = X7.

Par2

This parameter selects between the two digital clock signal sources (period 0 or period 1) and is important for digitally generated clock signals (Par0 = d).

Note

The external PLL is sourced by the PERIOD 0 or 1 signal depending on the board revision:

- syn1588® PCIe NIC Revision 2.0 PERIOD 0
- syn1588® PCIe NIC Revision 2.1 PERIOD 1

If a clock is generated by the PLL and another clock is generated digitally, the digital clock should use the PERIOD that is not used to source the PLL as signal source.

Par3

This parameter denotes the frequency that is to be generated and has to be in the following format:

Directly in Hz, e.g., 1536000 for a signal frequency of 1.536 MHz

Ending with ‘k’ to multiply the given value with 1000, e.g., 1536k for a signal frequency of 1.536 MHz

Ending with ‘M’ to multiply the given value with 1000000, e.g., 100M for a signal frequency of 100 MHz

Ending with ‘m’ to divide the given value by 1000, e.g., 250m for a signal frequency of 250 mHz

Ending with ‘u’ to divide the given value by 1000000, e.g., 50u for a signal frequency of 50 µHz

Par4

If the configuration for the PLL should be loaded from a file (provided by Oregano), this parameter determines the file name. This file has to be located in the same directory as the executable (ptpmmm or syn1588).

If the frequency command fails it will issue warning or error messages with additional information. For example, the command usage information (similar to the one given above) will be printed if a parameter is missing or wrong (e.g., a wrong format of the frequency parameter).

Examples

Note, the examples in this section are shown for a syn1588® PCIe NIC. For other syn1588® hardware restrictions may apply described in the previous chapters.

Case 1 – ptpmmm: Generate a low-jitter 100 kHz signal on X4

```
> frequency 001EC0FFFE85748B 1 0 p 4 1 100k
```

This command execution will configure the external PLL of the syn1588® device with the clock ID 001EC0FFFE85748B. The PERIOD from which the PLL is sourced depends on the hardware. Therefore, the period parameter (Par2) will be ignored and set internally to the correct PERIOD.

The generated signal will have a frequency of 100 kHz and will be routed to the SMA connector X4 of the NIC. The command response should be similar to the following:

```
INFO: Parsed parameters:
             Type:  PLL
             Source Period requested: Period1
             Output connector requested: XM4
  parsed desired Frequency:   100000.000000
INFO: Preparing PLL configuration!


INFO: Please wait a few seconds for the FPGA to prepare!
Waiting up to 30 seconds for jitter-cleaner PLL to lock!
....25....20..
INFO: Frequency config completed
>
```

Case 2 – ptpmmm: Digitally generate 1,536 MHz on X7

```
> frequency 001EC0FFFE85748B 1 0 d 7 0 1536k
```

This command execution will configure the FPGA period 0 of the syn1588® device with the clock ID 001EC0FFFE85748B. The generated signal will have a frequency of 1.536 MHz with a deviation of +24 mHz and will be routed to the SMA connector X7 of the NIC. The command response should be similar to the following:

```
INFO: Parsed parameters:
             Type:  Digitally generated
             Source Period requested: Period0
             Output connector requested: XM7
  parsed desired Frequency:   1536000.000000
INFO: Configure FPGA
INFO: FPGA configuration found for desired frequency: 1536000.000000!
INFO: Generated frequency will be: 1536000.024000 Hz

INFO: Please wait a few seconds for the FPGA to prepare!

INFO: Frequency config completed>
```

As can be seen from the command output, the generated frequency will differ slightly from the desired, due to the resolution of the period registers. If this frequency can also be generated by the external PLL (see Table 1) it is recommended to use the external PLL.

Case 3 – ptpmmm: Generate digital and low-jitter clocks in parallel

Case 1 and case 2 can be combined without problems to generate two clock signals, one sourced from the jitter-cleaner PLL and the other generated digitally.

Attention:

The PERIOD 0 or 1 register set (depending on the board revision as explained on page 10) should not be used for digital clock signal generation, if the PLL is already used to generate a clock signal. But the alternate PERIOD register set can still be used.

Case 4 – ptpmmm: Digitally generate two clock signals in parallel

```
> frequency 001EC0FFFE85748B 1 0 d 7 1 1536k
> frequency 001EC0FFFE85748B 1 0 d 4 0 2048k
```

These two commands will generate the clock digitally by configuring the period 1 register set to generate a 1.536 MHz signal and route it to connector X7 and the period 0 register set to generate a 2.048 MHz signal and route it to connector X4.

Case 5 – ptpmmm: Load a configuration file

For certain frequencies it might be necessary to contact Oregano for a PLL configuration. This is delivered via a configuration file and will be loaded automatically when **Par4** is equal to the filename. The file has to be moved/copied to the directory of the executable (ptpmmm or syn1588). For example, the file “hugo.dat” has been provided by Oregano and is to be used to configure the PLL.

```
> frequency 001EC0FFFE85748B 1 0 f 5 hugo.dat
```

This command will load the configuration from the file “hugo.dat”, which will set the jitter-cleaner PLL to generate the frequency specified in the file. The generated signal is routed to SMA connector X5.

Case 6 – syn1588

The command execution for the syn1588 command line tool is similar to the ptpmmm command line tool. The first three parameters are not needed for the syn1588 tool.

```
> frequency d 7 0 1536k
```

This command will yield the same result as the one described in Case 2.

Frequency input range

The jitter cleaner PLL equipped on a syn1588® PCIe NIC Revision 2.1 can be sourced by a signal provided on connector X8.

	Jitter-Cleaner PLL clock input frequency
Minimum	8 kHz
Minimum (recommended)	8 kHz
Maximum (recommended)	100 MHz
Maximum	710 MHz

Table 2:Frequency input range

“extclk” command

The “extclk” command is used to generate the internal clock from an input signal provided by the user at connector X8. The “extclk” command is only applicable for syn1588® PCIe NIC Revision 2.1 with the separately acquired ordering option “ExtClk” (see also: “AN002 - Application Note: Ordering syn1588® PCIe NIC”).

The “extclk” command can be used with the syn1588, the ptpmmm, and the fSync command line tools. While ptpmmm can be used to remotely configure syn1588® devices, the syn1588 and fSync tools configures only the local devices of the platform where it is executed.

The syntax is equivalent for both tools with a single exception. For ptpmmm three additional parameters are needed, denoting the target syn1588® PCIe NIC Rev 2.1 in the network.

Command syntax ptpmmm:

```
extclk [ptpmmm: clkid] [ptpmmm: port] [ptpmmm: domain]
          [Par0: p|f] [Par1: <frequency or filename>]
```

or for switching back to the internal clock:

```
extclk [ptpmmm: clkid] [ptpmmm: port] [ptpmmm: domain]
          off
```

Command syntax syn1588:

```
extclk [Par0: p|f] [Par1: <frequency or filename>]
```

or for switching back to the internal clock:

```
extclk off
```

Command syntax fSync:

```
./fsync [-d card] -m E -a Par1
```

or for switching back to the internal clock:

```
./fsync [-d card] -m I
```

Parameters

The extclk command can be executed with a single parameter = “off” to switch back to the internal clock.If the extclk command is to be used to switch to an external clock it has to be provided with the following two parameters:

Par0

This parameter selects between digitally generated (“d”) and external PLL generated clock signal (“p”), or loading a PLL configuration from a file (“f”).

Par1

This parameter denotes the frequency that is provided by the user at connector X8 and has to be in the following format:

Directly in Hz, e.g., 1536000 for a signal frequency of 1.536 MHz

Ending with ‘k’ to multiply the given value with 1000, e.g., 1536k for a signal frequency of 1.536 MHz

Ending with ‘M’ to multiply the given value with 1000000, e.g., 100M for a signal frequency of 100 MHz

If the extclk command fails it will issue warning or error messages with additional information. For example, the command usage information (similar to the one given above) will be printed if a parameter is missing or wrong (e.g., a wrong format of the frequency parameter).

Examples

Note, the examples in this section are shown for a syn1588® PCIe NIC. For other syn1588® hardware restrictions may apply described in the previous chapters.

Case 1 – ptpmmm: Use a 10 MHz input signal as source

```
> extclk 001EC0FFFE85748B 1 0 p 10M
```

This command execution will configure the jitter-cleaner PLL of the syn1588® PCIe NIC Revision 2.1 with the clock ID 001EC0FFFE85748B to use a 10 MHz signal provided by the user on connector X8.

The command response should be similar to the following:

```
Parsed parameters:
             calculated or from file :  Directly calculated
  parsed input Frequency:   10000000
Configure jitter-cleaner PLL
This can take several minutes!
Waiting up to 30 seconds for jitter-cleaner PLL to lock!
....25.....20.....
Successfully locked the external clock multiplexer,
the syn1588clock of the target NIC is now running on the clock provided by the
jitter-cleaner PLL!

Frequency config completed
>
```

Case 2 – ptpmmm: Switch back to the internal clock

```
> extclk 001EC0FFFE85748B 1 0 off
```

This command will set the target device back to the internal clock.

The command response should be similar to the following:

```
User requested that the external clock should no longer be used!

Successfully locked the external clock multiplexer,
 the syn1588clock of the target NIC is no longer running on an external clock!

Frequency config completed
>
```

Case 2 – ptpmmm: Load a configuration file

For certain frequencies it might be necessary to contact Oregano for a PLL configuration. This is delivered via a configuration file and will be loaded automatically when **Par1** is equal to the filename. The file has to be moved/copied to the directory of the executable (ptpmmm or syn1588). For example, the file “hugo.dat” has been provided by Oregano and is to be used to configure the PLL.

```
> extclk 001EC0FFFE85748B 1 0 f hugo.dat
```

This command will load the configuration from the file “hugo.dat”, which will set the jitter-cleaner PLL to the configuration as specified in the file.

Attention:

You have to provide the correct input frequency on connector X8 for this to work!

Case 3 – syn1588

The command execution for the syn1588 command line tool is similar to the ptpmmm command line tool. The first three parameters are not needed for the syn1588 tool.

```
> extclk p 10M
```

This command will yield the same result as the one described in Case 1.

Case 4 – fSync

This command will yield the same result as the one described in Case 1.

```
sudo ./fSync -d 0 -m E -a 10MHz
2023-10-05 12:05:39.725167 [INFO    ] [              ]  syn1588(R) fSync -
frequency Synchronization Engine
2023-10-05 12:05:39.725268 [INFO    ] [              ]  Build date: 2023-10-
03T05:02:37 - v1.17-5-ge201e4a9e
2023-10-05 12:05:39.725292 [INFO    ] [              ]  Copyright (c) Oregano
Systems - Design & Consulting GesmbH 2005-2023
2023-10-05 12:05:39.725306 [INFO    ] [              ]  Confidential unpublished
data - All rights reserved
2023-10-05 12:05:39.725322 [INFO    ] [              ]  Command line: ./fSync -d 0
-m E -a 10MHz
2023-10-05 12:05:39.725383 [INFO    ] [              ]  Port 0: adding config "d" =
"0"
2023-10-05 12:05:39.725402 [INFO    ] [              ]  Port 0: adding config "m" =
"E"
2023-10-05 12:05:39.725418 [INFO    ] [              ]  Port 0: adding config "a" =
"10MHz"
2023-10-05 12:05:39.725467 [INFO    ] [syn1588       ]  Syn1588Ifc requires at
least:
2023-10-05 12:05:39.725483 [INFO    ] [syn1588       ]  - linux driver version 1.4-
15-g05b7283
2023-10-05 12:05:39.725498 [INFO    ] [syn1588       ]  - windows driver version
10/05/2017, 10.9.16.182
2023-10-05 12:05:39.725554 [INFO    ] [syn1588       ]  Device /dev/syncD0 found
2023-10-05 12:05:39.725649 [INFO    ] [syn1588       ]  syn1588(R) PCIe NIC
Revision 2.1, Build 999 with HQ Oscillator
2023-10-05 12:05:39.725680 [ERROR   ] [syn1588       ]  Failed to read syn1588 PCIe
NIC board revision
2023-10-05 12:05:39.725727 [ERROR   ] [syn1588       ]  Failed to read board serial
number
2023-10-05 12:05:44.732545 [INFO    ] [              ]  fSync Engine is in External
Clock mode
2023-10-05 12:05:44.732602 [INFO    ] [              ]  Trying to enable pll with
frequency: 10000000 Hz
2023-10-05 12:05:44.797566 [ERROR   ] [syn1588       ]  Syn1588Impl: readReg error:
0x00002000
2023-10-05 12:05:44.802890 [ERROR   ] [syn1588       ]  Syn1588Impl: readReg error:
0x00002000
2023-10-05 12:05:44.804085 [ERROR   ] [syn1588       ]  Syn1588Impl: readReg error:
0x00002004
2023-10-05 12:05:44.805223 [CRITICAL] [              ]  No PLL of type "Si5328" has
been found!
2023-10-05 12:05:44.805265 [INFO    ] [              ]  Waiting for PLL to lock
(max. 30 times)
.2023-10-05 12:05:44.805294 [INFO    ] [              ]  PLL locked!
2023-10-05 12:05:44.805325 [INFO    ] [              ]  Auxiliary reg: 0x00000000
2023-10-05 12:05:44.805352 [INFO    ] [              ]  Waiting for FPGA clockmux
to lock (max. 10 times)
2023-10-05 12:05:53.809283 [CRITICAL] [              ]  External clock multiplexer
did not lock in time!
2023-10-05 12:05:53.809342 [INFO    ] [              ]  Waiting for PLL to lock
(max. 1 times)
.2023-10-05 12:05:53.809373 [INFO    ] [              ]  PLL locked!
2023-10-05 12:05:53.809403 [INFO    ] [              ]  Auxiliary reg: 0x00000001
2023-10-05 12:05:53.809437 [INFO    ] [              ]  Waiting for FPGA clockmux
to lock (max. 1 times)
2023-10-05 12:05:55.810336 [INFO    ] [              ]  Waiting for PLL to lock
(max. 1 times)
.2023-10-05 12:05:55.810406 [INFO    ] [              ]  PLL locked!
2023-10-05 12:05:55.810444 [INFO    ] [              ]  Auxiliary reg: 0x00000001
2023-10-05 12:05:55.810472 [INFO    ] [              ]  Waiting for FPGA clockmux
to lock (max. 1 times)
```

```
2023-10-05 12:05:57.811343 [INFO    ] [              ]  Waiting for PLL to lock
(max. 1 times)
.2023-10-05 12:05:57.811411 [INFO    ] [              ]  PLL locked!
```

Note, that unless syn1588 or ptpmmm, fSync continuously monitors the locked-state of the PLL. If the external clock signal is removed or missing, the user will be notified.

Frequently Asked Questions

The ptpmmm command line tool issues a “Wrong PLL ID” error message when executing the frequency command

This can occur if the target syn1588® device has an older firmware version (as listed in the Requirements) or if the syn1588® device is not equipped with an external jitter cleaner PLL.

The frequency command does not finish its execution

This can occur if an unusual output frequency is to be generated by the PLL. The frequency command will try to find a valid PLL configuration which can take several minutes for certain frequencies.

No valid PLL configuration can be found

The frequency command returns a warning that no valid PLL configuration is possible for the desired frequency, if the calculation was unsuccessful. If a specific frequency is needed but cannot generated, Contact Oregano Systems with details regarding the desired frequency. It is possible to add specific frequency settings with configuration files.

The calculated FPGA generated frequency differs from the requested frequency

The period settings of the FPGA provide a wide range of possible values, but do not cover the full range of frequencies. The calculation run by the syn1588 or ptpmmm tool will always give feedback of the requested vs. the actual possible frequency that will be generated. Most of the time these will match, but if they differ, it is recommended to generate the desired frequency by the external PLL (if available and it is in the frequency range of the PLL). If this is not possible, Contact Oregano Systems for an alternative solution.

How to add a specific frequency configuration file

The file received from Oregano has to be moved/copied into the same directory as the syn1588 or ptpmmm tool. It is selected by supplying “f” as Par0 and the filename as frequency parameter (Par3) to the frequency command (see Case 5).

When executing the “frequency” command the ptpmmm or syn1588 utility issues the following error message: “The jitter-cleaner PLL is already in use as source for the syn1588core, you have to provide a different source for the syn1588core before you can use the jitter-cleaner PLL!”

This happens if the syn1588® PCIe NIC Revision 2.1 has been configured to generate its internal clock via a signal provided on the connector X8 (i.e., the “extclk” command as explained in this application note).

Before the jitter-cleaner PLL can be used to generate an output frequency via the “frequency” command, the syn1588® PCIe NIC Revision 2.1 has to be switched back to a different clock source → go to the “extclk” command section of this application note for information about turning this off (i.e., extclk off).

Oregano Systems
A Meinberg Company

Franzosengraben 8
A-1030 Vienna
Austria
http://oregano.at
contact@oregano.at

Oregano Systems does not assume any liability arising out of the application or use of any product described or shown herein nor does it convey any license under its patents, copyrights, or any rights of others.

Licenses or any other rights such as, but not limited to, patents, utility models, trademarks or tradenames, are neither granted nor conveyed by this document, nor does this document constitute any obligation of the disclosing party to grant or convey such rights to the receiving party.

Oregano Systems reserves the right to make changes, at any time without notice, in order to improve reliability, function or design. Oregano Systems will not assume responsibility for the use of any circuitry described herein.