RF2TTC and QPLL behavior during interruption or switch of the RF-BC source

Study to adapt the BC source choice in RF2TTC during interruption of the RF timing signals

Contents

I. Introduction

The RF2TTC modules (receiving the RF signals in experiments) provide 4 different clock sources to the experiments:

- BCint: an internal, standalone clock, frequency ≈ 40.0785MHz
- BC1 and BC2: Bunch Clocks respectively driving beam1 and beam2. Their frequencies increase during ramping, and are then slowly shifted to be phase aligned to the BCref (see below). These signals disappear during 1ms for resynchronization before each 10h run, during beam setup mode (about 1hour before ramping).
- BCref: stable Bunch Clock signal during the run. This signal is the final bunch clock to which BC1 and BC2 will be locked and phase aligned at the flat top. This signal disappears at the same time as BC1 and BC2 during 1ms for resynchronization, during beam setup mode (about 1hour before ramping).

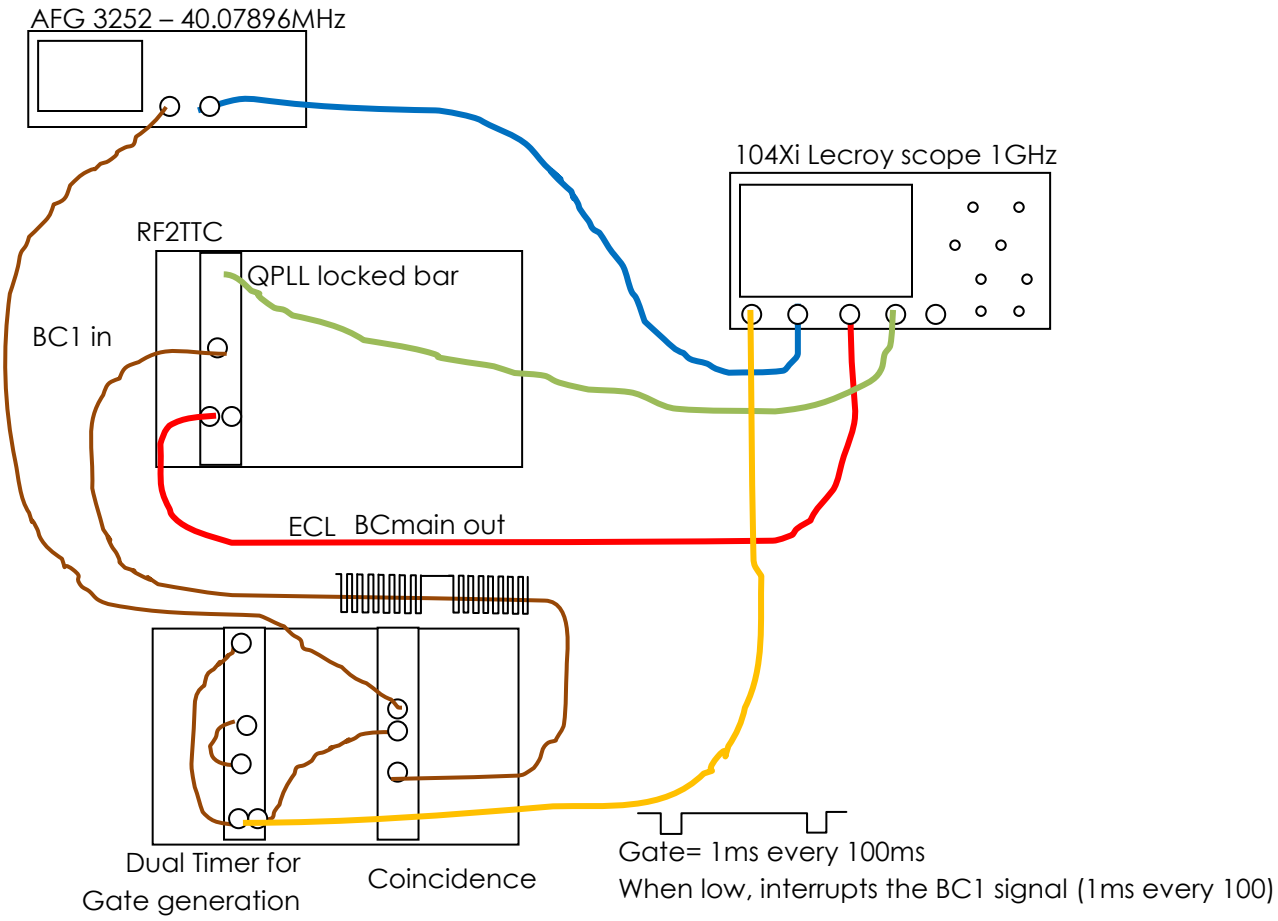
A study on the way for the experiments to handle this gap of 1ms without Bunch Clocks signals from the RF has been done. As the 'BCmain' output of the RF2TTC can be chosen between BCint, BC1, BC2 and BCref, this study is based on QPLL characteristics and on measurements of the BCmain signal out of the RF2TTC (and therefore out of the QPLL) in several conditions:

- Simple BC source interruption of 1ms at the input of the RF2TTC with the QPLL in the modes 1 or 0: autorestart=1 (automatically scans the frequency range in case of loss of lock) and autorestart=0 (scans the frequency range only after a reset)
- Switch between internal and external BC sources at the input of the RF2TTC with the QPLL in the modes 1 or 0
- mode transition of the QPLL between mode 1 and mode 0

The conclusion of this study, with a proposal of several solutions to smooth over the clock transition periods, is presented at the end of this document.

II. QPLL behavior during 1ms BC interruption

A. Setup



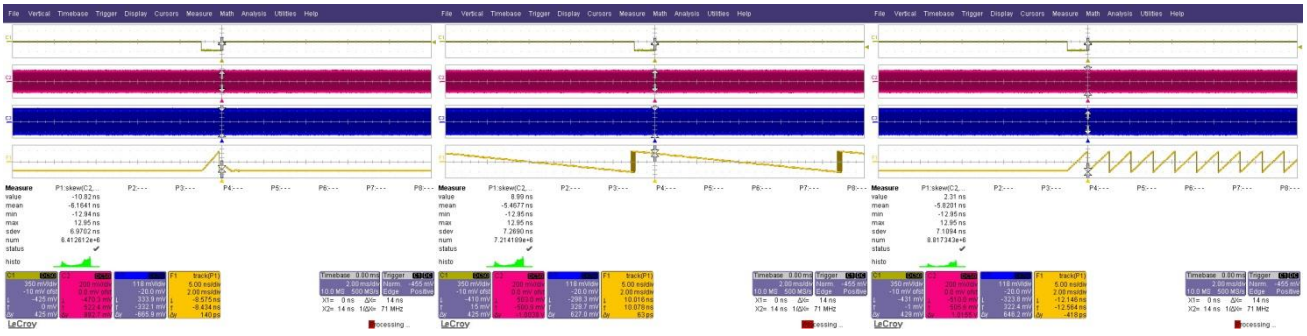
The scope is triggered by the gate signal (CH1)
QPLL locked signal is monitored using a passive probe on the PCB (CH4)
F1 displays the skew between external clock (CH2) and BCmain out (CH3)
BC1 source is set as external clock (reg 7FBFC=1), BCmain as 'BC1'(reg 7FB8C=3)

B. Auto-restart mode (QPLL mode 1)

RF2TTC register 7fb80 set to 1, default configuration

Reminder: in autorestart mode, as soon as the QPLL detects a jump in the phase bigger than π/4, it is declared as 'unlocked' and it starts a full scan of its digital range. [2]

First measurements, various cases overview

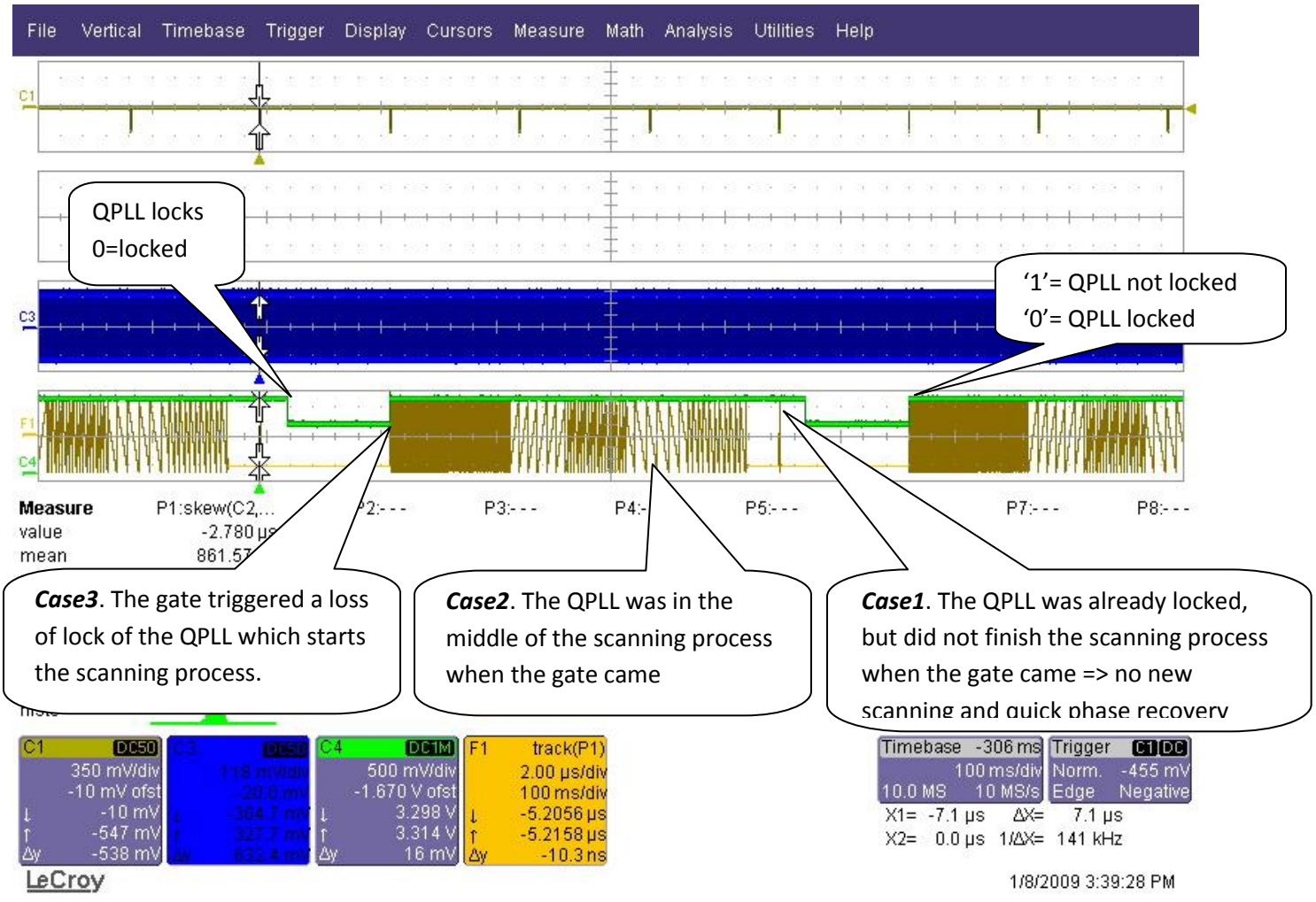


Gate (no BC1 when gate is low)

Phase. (Flat waveform means QPLL locked)

Case 1: quick phase recovery ***Case 2***: scanning already running ***Case 3***: scanning started by the gate

Second measurements, QPLL lock signal measured, BC high during interruption



Case3. The gate triggered a loss of lock of the QPLL which starts the scanning process.

Case2. The QPLL was in the middle of the scanning process when the gate came

Case1. The QPLL was already locked, but did not finish the scanning process when the gate came => no new scanning and quick phase recovery

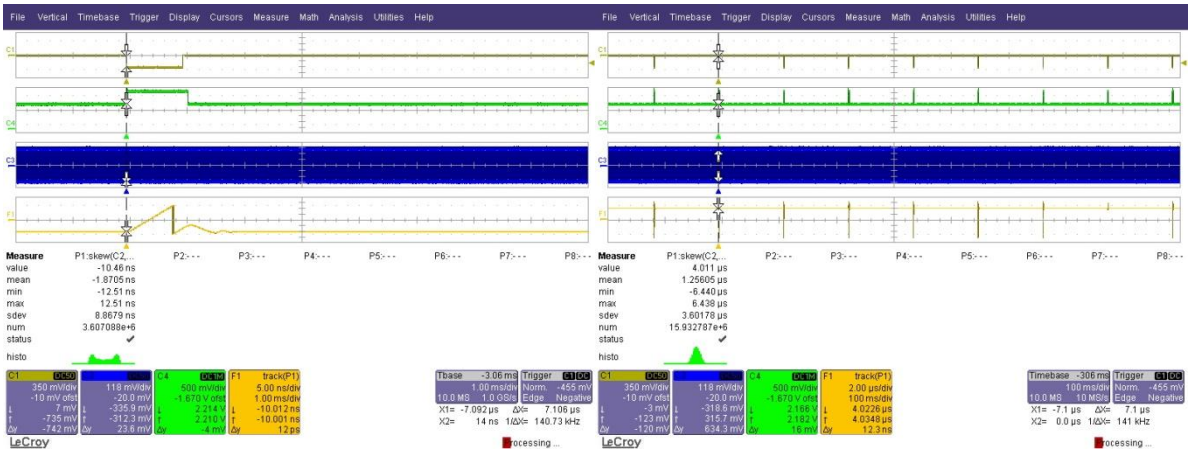
Third measurements: QPLL lock signal measured, BC low during interruption

Same behavioral. The only difference observed is an expected change in the phase sign (F1).

C. Relock-after-reset mode (QPLL mode 0)

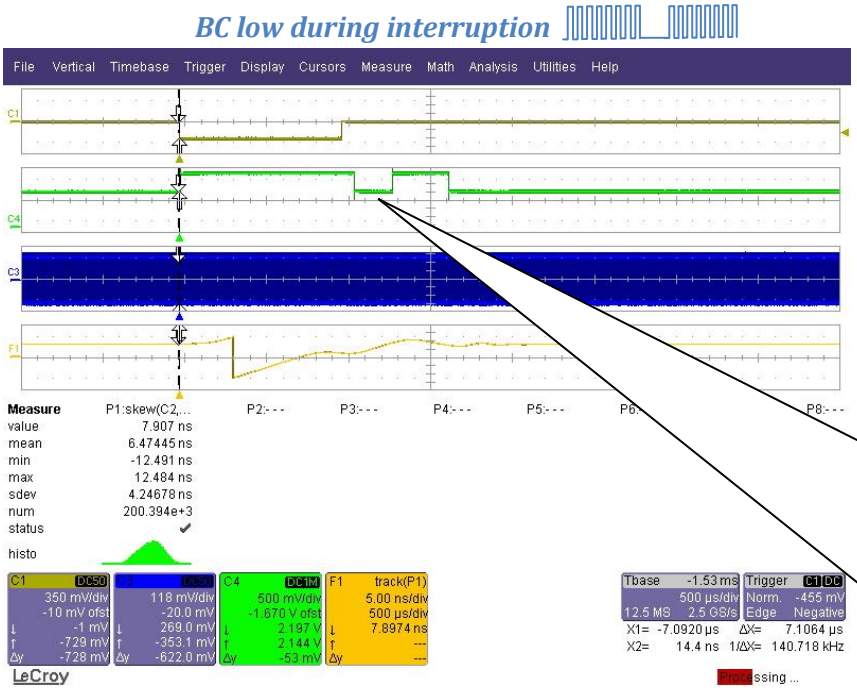
RF2TTC register 7fb80 set to 0

BC high during interruption



The QPLL is unlocked during the gate, its output (BCmain) drifts during the duration of the QPLL being unlocked, but no scanning is initiated. As soon as the signal is back, the QPLL finds the lock back without strong variations.

BC low during interruption



At this point, the QPLL output frequency is close enough to the input frequency so that the phase stays below π/4 during more than 1000 clock cycles. The QPLL declares thus that it is locked, although the 2 frequencies are still slowly drifting. 300µs later, the 2 frequencies have drifted more than π/4. The QPLL declares itself as 'unlocked' again, and finishes the locking process. This is a normal behavior.

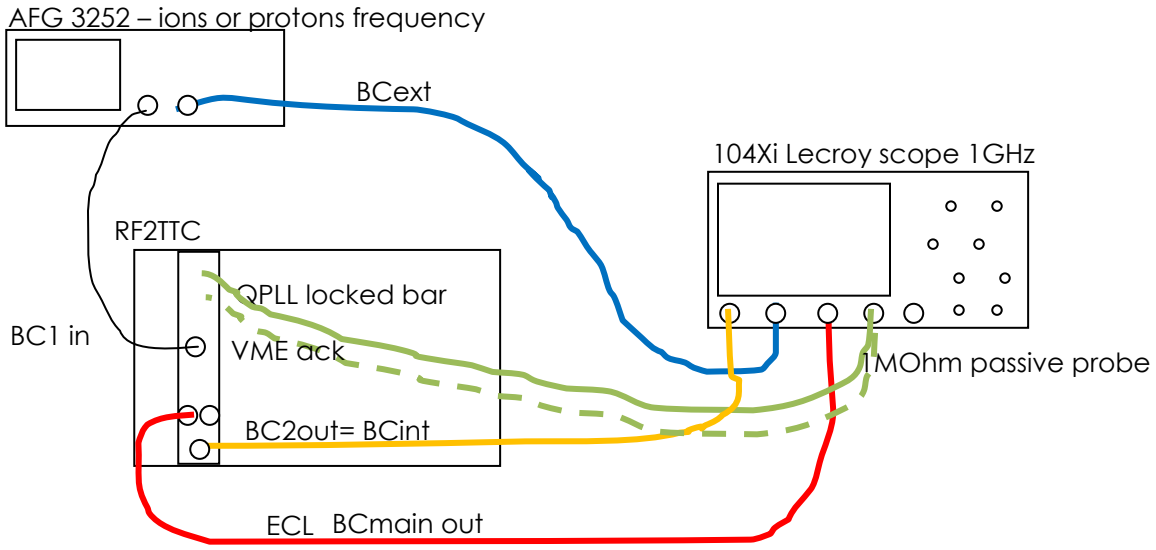
III. QPLL behavior during BC source switch (BCint used during BCext interruption)

A solution to still have a clock during the resynchronization period of 1ms would be to switch to internal Bunch Clock (BCint) between beam dump and end of setup mode, and then switch back to the required external Bunch Clock (BCext) (for example BC1). The effect of this switch is studied for the QPLL in autorestart mode (1) and in relock-after-reset mode (0).

The values of the above-mentioned BC frequencies are the following:

- BCint= theoretical value of the oscillator 80.1574MHz +/-25ppm => range of [40.0777MHz; 40.0797MHz]. Sample used for the test: 40.0783MHz.
- BCext at 450GeV= frequency ***40.0784187MHz*** for 450GeV ions, ***40.07887834MHz*** for 450GeV protons

A. Setup



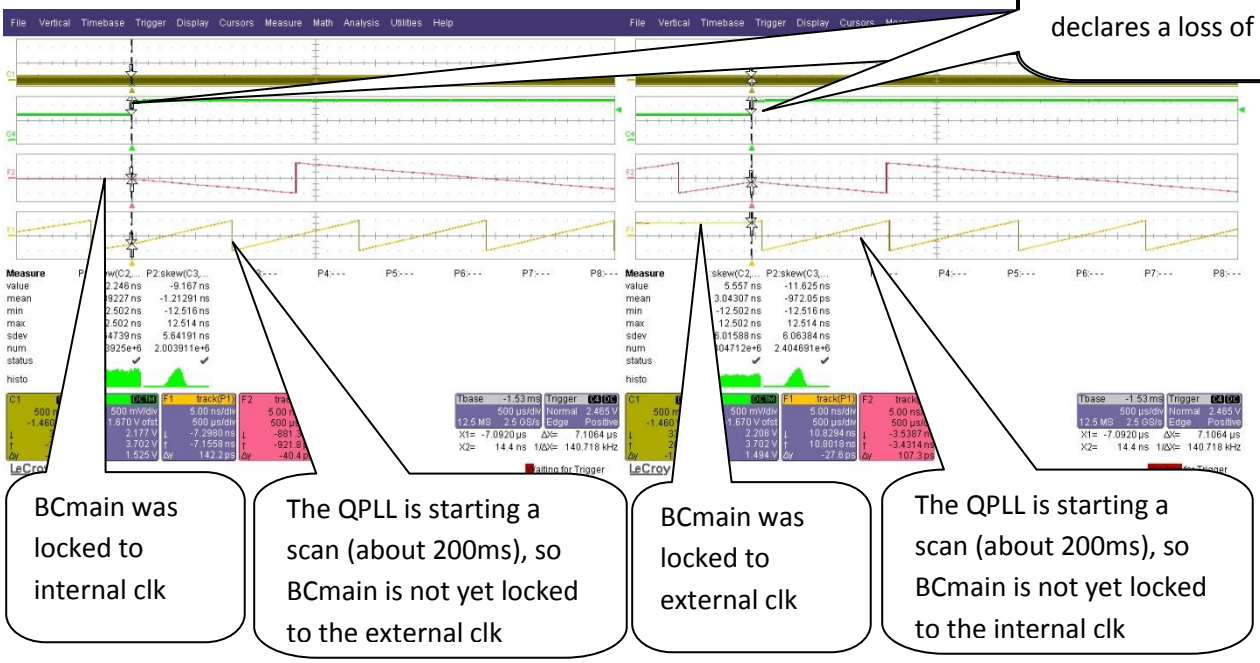
The scope was triggered by the vme ack signal. As I saw that only about 20% of the switch actions were generating loss of lock (and, therefore, interesting curves), I then changed to a trigger on the QPLL lock rising edge.
RF2TTC configuration: BC1 source is set as external clock (reg 7FBFC=1), BCmain is controlled by VME and changes from BCint (reg 7FB8C=0) to 'BC1' (reg 7FB8C=3).
CH1: BC2out= BCint
CH2: BCext (generator)
CH3: BCmain out
CH4: / QPLL lock
F1 (yellow) displays the skew between external clock and BCmain out
F2 (pink) displays the skew between internal clock and BCmain out.
Thus, a flat line on F1 means that BCmain out (=QPLL out) is locked to the external clock, a flat line on F2 means that BCmain out is locked to the internal BC. A flat line in neither F1 nor F2 means that the QPLL is scanning.

B. Auto-restart mode (QPLL mode 1)

From 450 GeV Protons frequency (40.07887834MHz) to BCint and back

Internal to external

External to internal

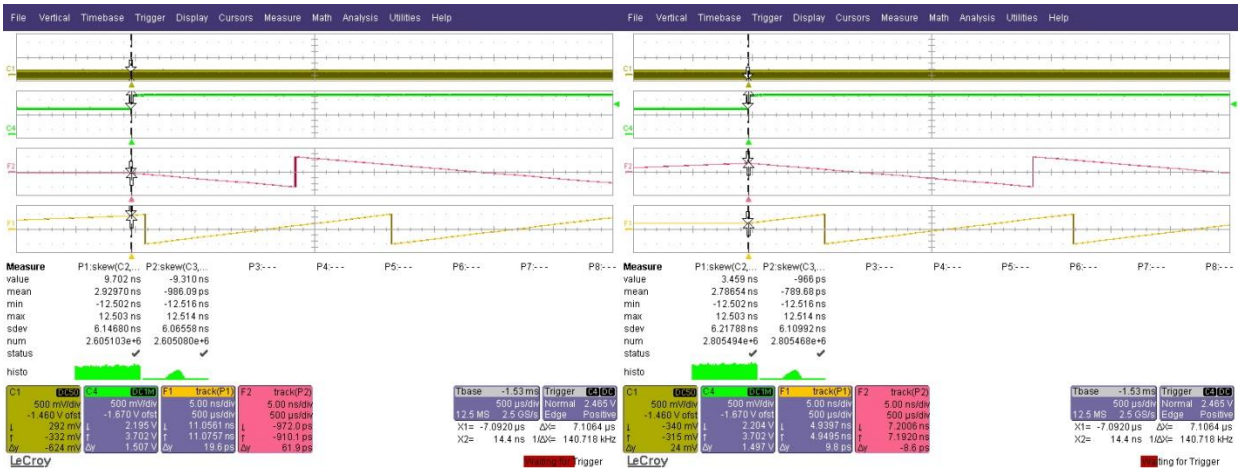


From 450 GeV Ions frequency (40.0784187MHz) to BCint and back

⇨ no difference between ions and protons

Internal to external

External to internal

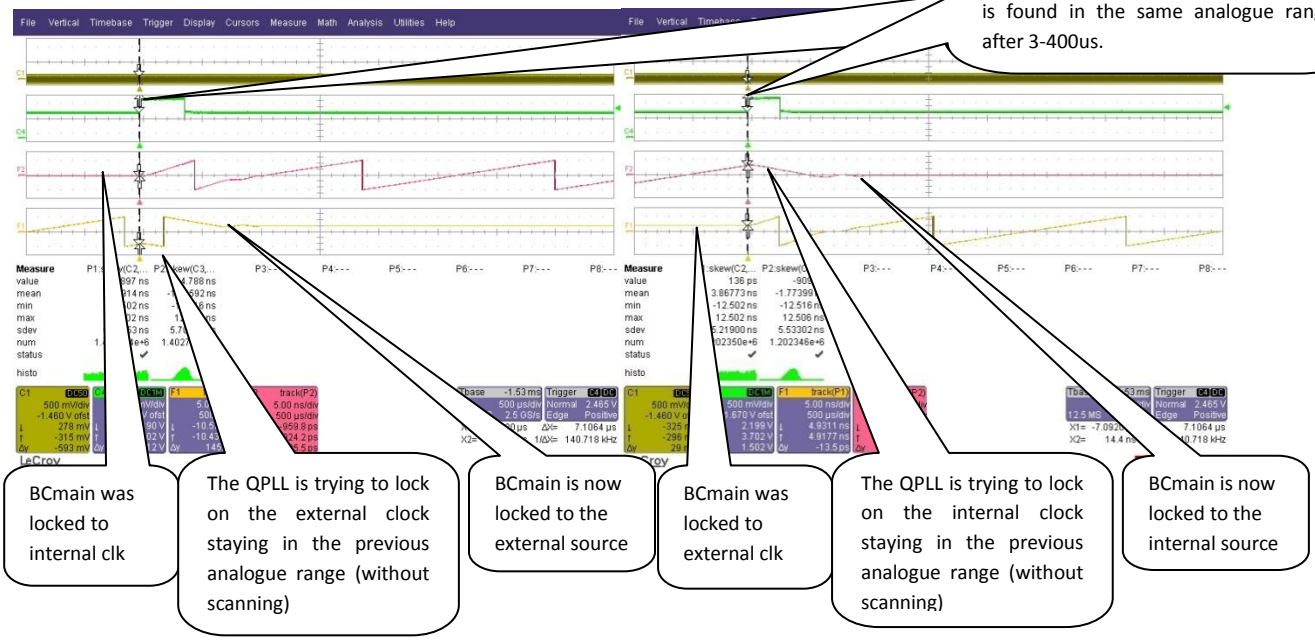


C. Lock-after-reset mode (QPLL mode 0)

From 450 GeV Protons frequency (40.07887834MHz) to BCint and back

Internal to external

External to internal

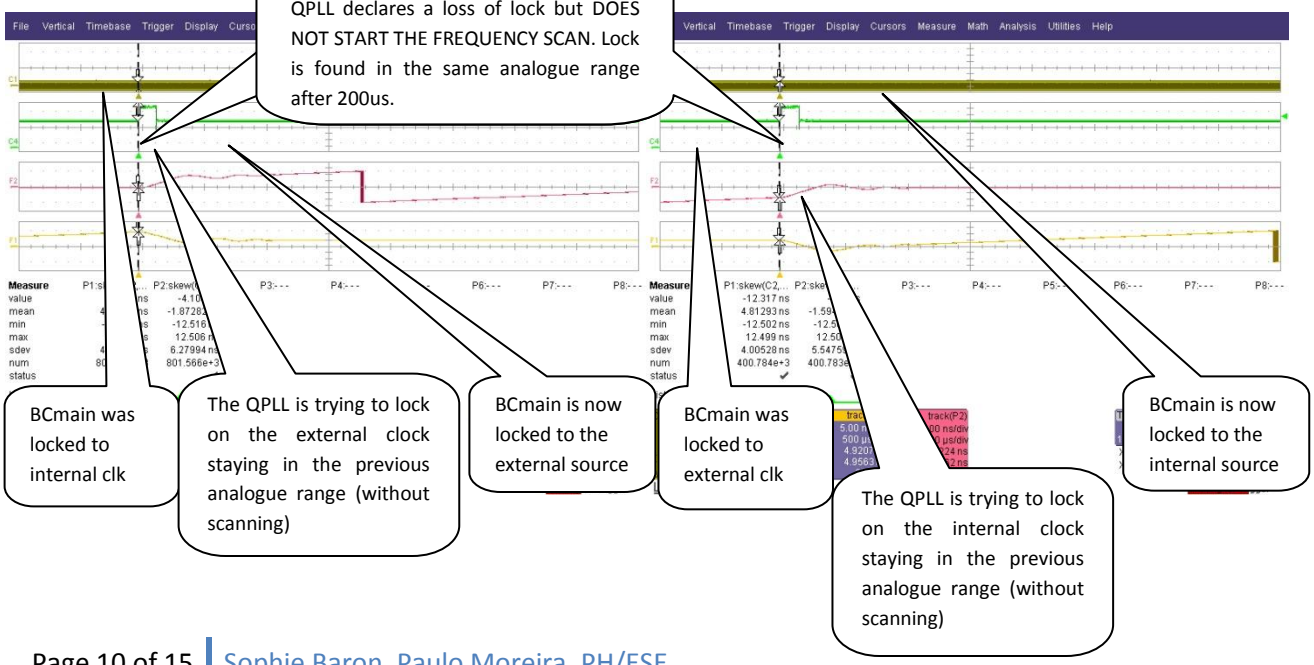


From 450 GeV Ions frequency (40.0784187MHz) to BCint and back

⇨ the jump is smaller than with the protons, the locking time is shorter

Internal to external

External to internal



IV. QPLL behavior during change of mode (0 to 1 and vice versa)

If locked, the QPLL can be switched from autorestart to relock-after-reset mode and back without any scan process being initialized:

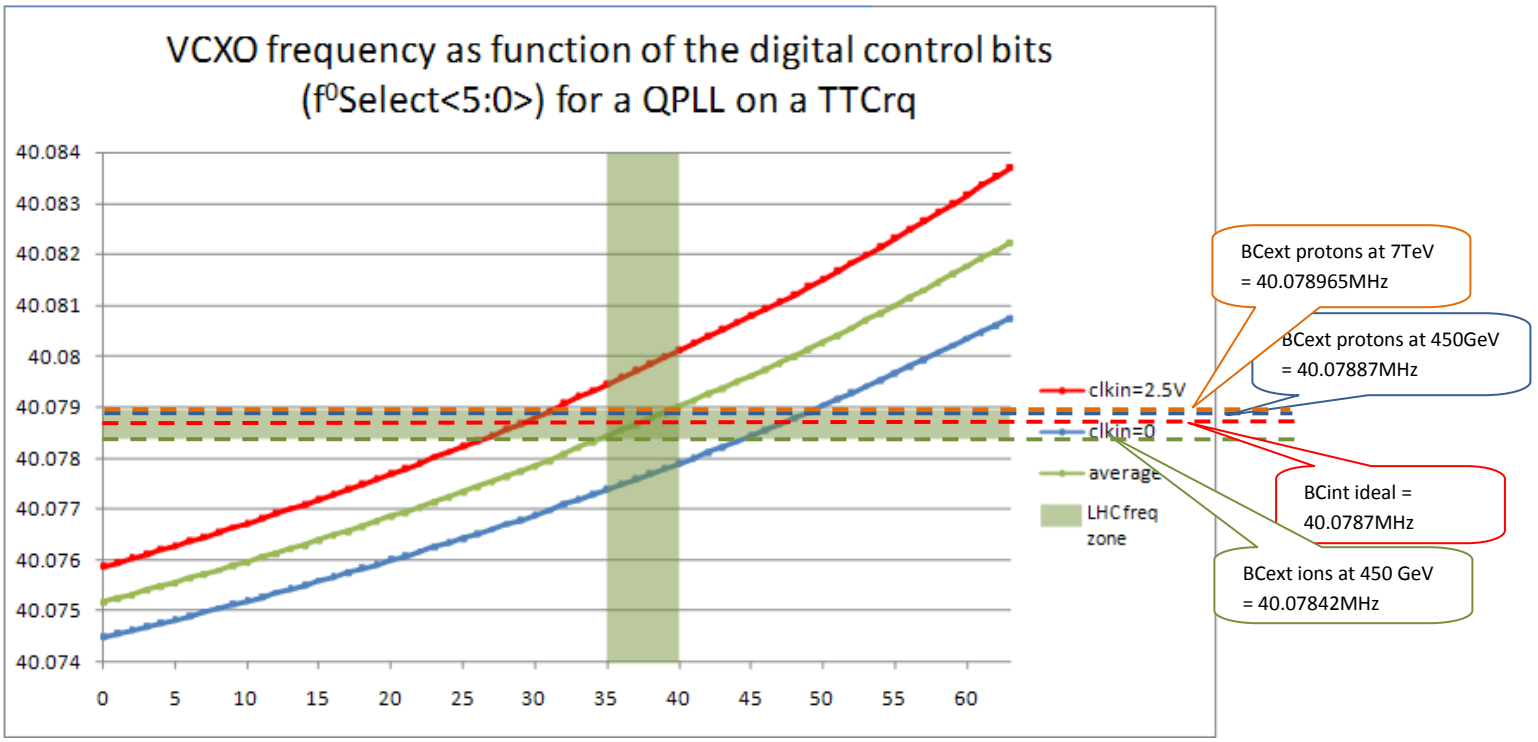
The QPLL does not lose the lock or starts a scan when a transition between autorestart mode and relock-after-reset mode occurs

V. Conclusions

The RF frequencies will be shortly interrupted (1ms) during the machine setup (1hour before ramping). The timing distribution in experiments is often made of cascade of QPLL chips, which may respond to this interruption by generating a chain reaction of unlocks, propagating loss of synchronization in the DAQ systems and requiring a substantial delay to recover.

Two solutions are proposed for the experiments to minimize the impact of this event on the front-end and DAQ of the detectors. They are based on the following statements:

- As much as possible, softening the BC transitions would require avoiding the frequency scan proposed by the QPLL at each loss of lock in autorestart mode
- All the commonly used frequencies are located between ***40.0784187MHz*** (ions @ 450GeV) and ***40.07896474MHz*** (protons flat top @ 7TeV), that we will call *LHCrange* = [40.0784187MHz; 40.07896474MHz].
- The full LHCrange fits perfectly well within each of minimum 10 analogue ranges* (between 32 and 42 in the following figure, representing the locking range analysis of the TTCrq).



Conclusion 1: which locking mode for the QPLL on the RF2TTC?

As the full LHCrange fits perfectly well within each of about 10 analogue ranges (third statement above), the QPLL will never need to change its analogue range during normal conditions ***if it has been correctly initialized***. The only exception would be if a scan of the full digital range occurs (in case of a reset or a loss-of-lock in autorestart mode). In that case, it is possible that the new chosen analogue range differs from the previous one.

Working in the mode 0 (rescans the frequency range only when a reset occurs) would force the QPLL to stay in the analogue range chosen during the latest reset. This has the disadvantage of dividing the total locking range by a factor of about 3.5 (the digital range is about 7kHz, the analogue range is about 2kHz). However, this is not a problem as, as it has been noted previously, all the frequencies are anyway fully represented in each of several analogue ranges.

We propose therefore to work with the QPLL of the RF2TTC in the so-called mode 0. As soon as the frequency of the input signal stays between 40.0780 MHz* and 40.0791 MHz*, this mode should not prevent the QPLL from locking precisely to the input frequency, but it will avoid a brutal scan if the input clock disappears or changes its source.

However, users have to be sure that the analogue range chosen by the QPLL is one of right ones (in our example on the figure above, between 35 and 40 to be very comfortable), by resetting the QPLL with a significant BC source at its input (ideally, with and input frequency between 40.0784 MHz and 40.0789).

The RF2TTC internal frequency is usually a good candidate (nominal 40.0787MHz), but the manufacturing tolerances could push the nominal frequency out of the ideal range**. Another good candidate would be the BCref frequency, available all the time except during maintenance and during the 1ms resynchronization gap. Its value, set to the flat top frequency of the particles used in the LHC, is 40.078965 MHz, which should guaranty the selection of a reasonable analogue range of the QPLL (although a little high).

These remarks are also valid for the other boards containing QPLLs receiving the clock from the RF2TTC.

***These values are theoretical. They have been verified on the RF2TTCs and on TTCrqs, but may vary with the way the QPLL are implemented on other types of PCBs. Usually the values tend to be lower for boards which do not respect the QPLL layout recommendation to minimize parasitic capacitance (the entire curves are shifted down with respect to the figure below).**

**** MCO-1S3-PE-p6 from QuartzCom. Nominal freq 80.1574MHz +/-25ppm (resulting nominal frequency=40.0787MHz, with a range of = [40.0777; 40.0797]). The frequency can be measured on your RF2TTC, and the oscillator can be exchanged if it is too far from the ideal range.**

Conclusion 2: switching or interrupting input clock source during resynchronisation?

The tests previously presented show that both solutions are possible with QPLL in relock-after-reset mode (mode 0).

- BC interruption: if the BC source is kept as one of the external BCs, there will be a period theoretically with no signal at the QPLL input during the RF resync. As seen during the test, the frequency of the clock provided by the QPLL will slowly drift to one of the ends of the analogue range, waiting the input signal to be back. A few 100s of µs after the signal being back at the input, the output clock will smoothly lock back to the input. The presence of some noise that could be interpreted by the QPLL as a frequency will not significantly change this behavior: the frequency of the clock provided by the QPLL during this time may increase and decrease instead of evolving in the same direction, but it will always stay in the same analogue range.
- Switch between internal and external: the experiment could chose to switch from external clock to internal clock after the beam dump, and come back to external clock after the setup mode. In that case, a short period of 200 to 400µs will occur at each transition, during which the QPLL will smoothly change its frequency to lock to the new signal (which is still on the same analogue range). It will set its flag to 'lock is lost' but will not scan the full frequency range.

Conclusion 3: QPLLs initialization precautions

To avoid any cascade of QPLLs losing locks in experiments during events in the clock sources, we advise the users to work with the QPLL of the RF2TTC in mode 0 (see conclusion 1).

A minimum configuration would be to set the QPLLs of the RF2TTC (top of the QPLL chain) on the mode 0, and keep the other QPLLs in mode 1. This will guaranty a smooth behaviour of the clock distributed within the experiment during bunch clock disturbances. Moreover, it strongly reduces (but does not cancel) the chances of having a QPLL deeper in the detector losing the lock and beginning a scan.

An even more conservative solution would be to set all the QPLLs possible in mode 0.

However, the mandatory condition to this solution is to ensure that ***all the QPLLs set in mode 0 are reset after power up***

- ***Sequentially: first the QPLL placed on the top of the time distribution tree, then the qplls on the second level, and so on.***
- ***With a significant clock (ideally with a frequency higher than 40.0785 MHz) at the input of the QPLL placed at the top (usually on the RF2TTC) to guaranty the choice of the best analogue range. Check the internal frequency of your RF2TTC to ensure it is a good candidate for QPLL setting (there is a possibility to exchange the crystal if its frequency is not high enough), or use the frequencies provided by the RF out of the resynchronization period.***

VI. References:

[1]: QPLL Manual, P. Moreira: http://proj-qpll.web.cern.ch/proj-qpll/images/qpllManual.pdf

[2]: QPLL locking mechanism, P. Moreira, S. Baron: http://proj-qpll.web.cern.ch/proj-qpll/images/qpllLockingMechanismNote.pdf

[3]: TTC web page: http://ttc.web.cern.ch/TTC/