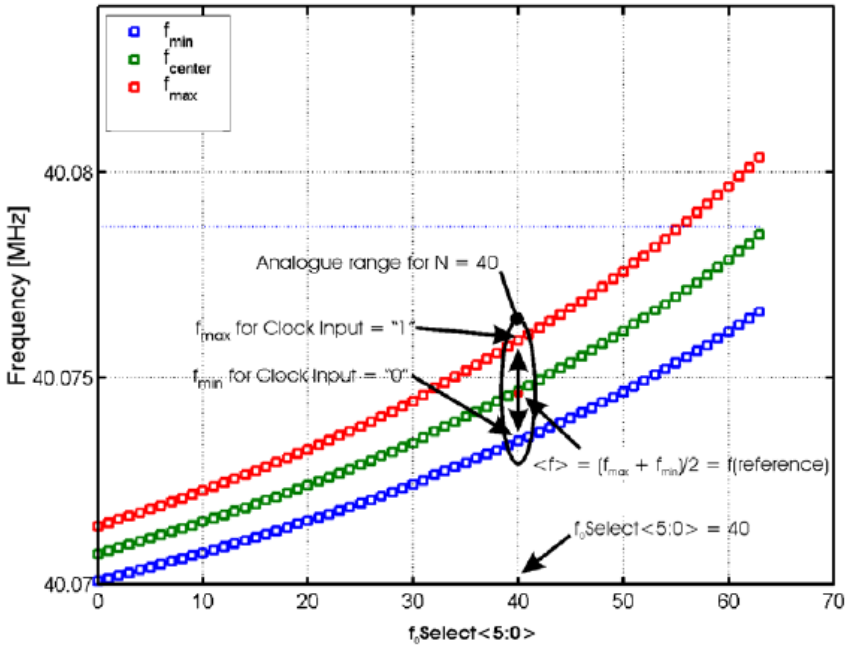
QPLL locking mechanism

A brief note to explain the QPLL locking mechanism and the 'locked' signal behaviour.

Introduction: The QPLL chip provides an indication of its locked status on the "locked" signal (pin #9). The meaning of this flag, as well as its behaviour, is explained below.

QPLL locking ranges: The QPLL has a full locking range of Δ ≈ ±3.7 kHz around f = 40.0786 MHz. However, this locking range (named 'digital range') is made of 64 'analogue ranges' (varying Δ between ≈ ±0.8 kHz and ≈ ±2 kHz) , materialized on the following graph as vertical segments between blue and red squares for a given value of $f_0$select<5:0>.



Operation modes:

1. External mode (externalControl = "1", pin #4)

The QPLL will deliver a frequency which depends on 2 parameters:

- The $f_0$select<5:0> bits
  - Determine its running analogue range (the vertical segment defined above)
- The output of the phase detector (between the internal 40MHz clock and the input clock signal):
  - When a clock signal is fed to the QPLL and its frequency is within the frequency range of the selected analogue range then the QPLL will lock to the reference.
  - If the input clock frequency is outside the selected analogue range, the QPLL output frequency is unpredictable and can vary with time. But it will be somewhere between fmin and fmax.
  - It is also possible to feed the QPLL input with DC levels. In this case:
    - If the input level is '0', the QPLL output frequency will be the minimum frequency of the selected analogue range fmin (blue square);
    - If the input level is '1', the QPLL output frequency will be the maximum frequency of the selected analogue range fmax (red square).

  ***Note**: as described above, even if the input clock is theoretically not required for the QPLL in external (or standalone) mode, its behaviour has an impact on the delivered frequency.*

2. Internal mode (externalControl = “0”, pin #4)

During the initial phase of the locking process, the QPLL does a binary search of the *digital* range (i.e. the 64 analogue ranges) and chooses the best $f_0$select<5:0> value to centre the analogue range around the clock input. Later, if it happens that the input clock **frequency** drifts or changes and goes out of the *analogue* range, the QPLL will behave as follows:

- If it is in the “auto restart mode” (autoRestart = “1”, pin #5) it will clear the locked flag (“locked” signal) and it will re-launch the locking procedure. This operation lasts about 180ms.
- If the “auto restart mode” is off (autoRestart = “0”, pin #5), the output locked flag will be cleared, and the output frequency will drift to the closest end of its current analogue range (either the max or the min) and will stay there until the input frequency goes back inside the analogue locking range (it would then last about 250us), or until a reset is applied which will re-launch a full phase locking procedure (180ms).

Lock detection and lock flag

: The locked output pin of the QPLL (pin #9, ‘1’=locked, ‘0’=loss of lock) is based on a comparison between the phase of the input clock and the phase of the internal clock (generated by the PLL). The result of this comparison is synchronized with the internal 40MHz clock. This internal signal is called the ‘instant lock’.

- In the external mode, the ‘locked’ output pin behaves as the internal ‘instant lock’ signal;
- In the internal mode, the ‘instant lock’ signal feeds a state machine which basically adds some hysteresis to the ‘locked’ output pin value (see ‘lock acquisition and tracking section’)

Lock acquisition and tracking:

- If the QPLL is initially unlocked, it waits for 1000 consecutive ‘1’ states of the instant lock signal before declaring itself as ‘locked’.
- If the QPLL is locked, it waits for 32 consecutive ‘0’ of the instant lock signal before declaring itself as ‘unlocked’

Conditions for phase tracking:

- If the input reference phase changes abruptly but less than $\pi/4$, the QPPL will maintain the ‘locked’ flag and will slowly tend to the new phase. If the abrupt change is bigger than $\pi/4$, then the lock flag will be cleared indicating a loss of lock.
- The QPLL will be able to track continuous phase variations, provided that the phase modulation bandwidth is less than 6 kHz.

Particular cases:

- The input signal is a constant “1”:
  - Then the lock flag is set as ‘LOCKED’ (even if the QPLL can’t physically lock on the signal). The QPLL won’t thus scan the frequency range, by will slowly drift and will deliver a frequency = fmax of its analogue range.
- The input signal is a constant “0”:
  - Then the lock flag is set as ‘UNLOCKED’ and the QPLL will restart the phase locking procedure all over again without stopping.