25th IEEE Real Time Conference
La Bioloda, Elba, Italy

Brookhaven National Laboratory

U.S. DEPARTMENT of ENERGY

SAMBA

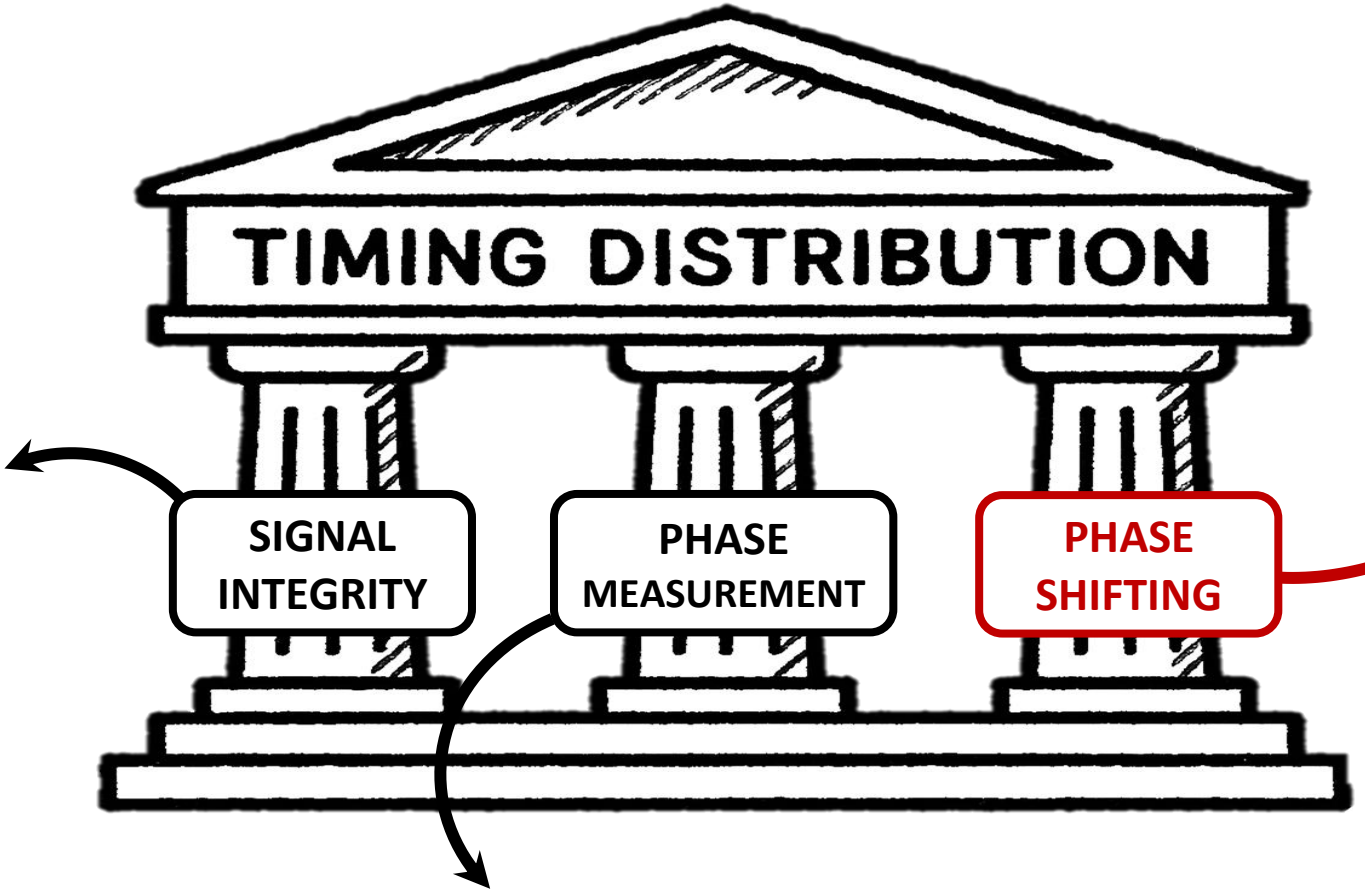
System-Agnostic Method for Biphasic Alignment

Maurício Féo, Hao Xu, Hucheng Chen, Shaochun Tang, Andy Liu, Eric Buschmann, Mayan Tamari, George Chatzianastasiou, and Marcos Oliveira

May 27th, 2026

@BrookhavenLab

The pillars of a timing distribution system



10's of femtoseconds (Si5395 ≅ 70 fs)

10's of femtoseconds (DDMTD ≅ 10 fs)
Limited by signal integrity

10's of picoseconds in most PLLs & FPGAs (Si5395 ≅ 70 ps)

>1.5 ps with AMD's proprietary Phase Interpolator

Delay lines: Sub-ps but limited range

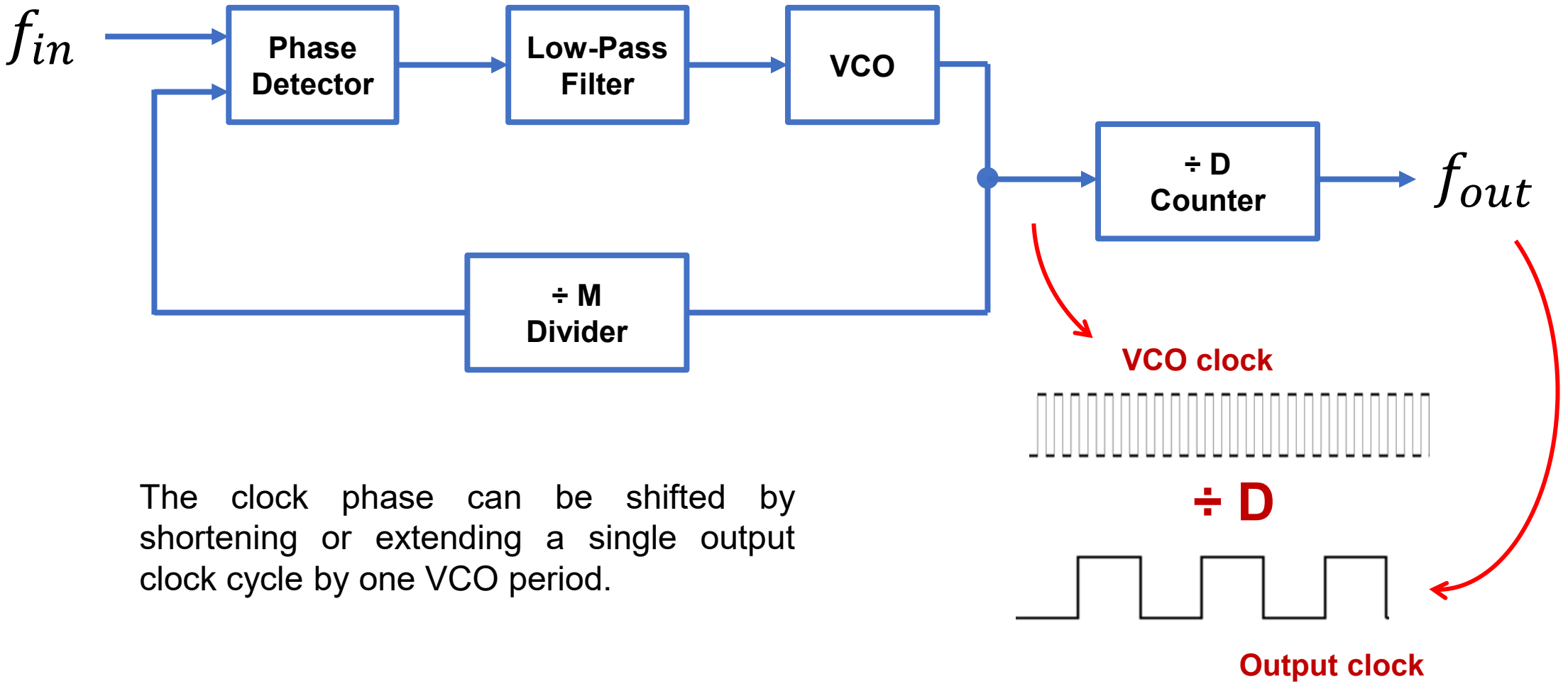
System-**A**gnostic **M**ethod for **B**iphasic **A**lignment

- **SAMBA** is a system-agnostic technique for improving the phase-shift resolution of clocking resources by **orders of magnitude**.

- **SAMBA** can be implemented **in FPGAs** or outside using **only COTS** components. It **does not require new hardware**.

- **The limiting factor is signal integrity.**

PLL-based Phase Shifting

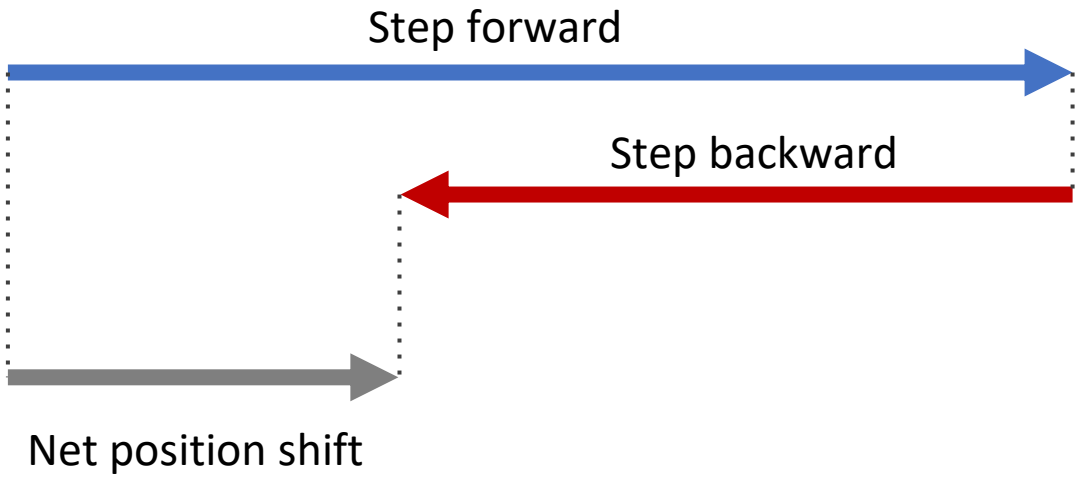


The clock phase can be shifted by shortening or extending a single output clock cycle by one VCO period.

Shifting your body with SAMBA

- **SAMBA** is also a dance characterized by steps forward and backward

- If you want to move forward in the samba parade, you adjust your forward steps to be longer than the backward steps:

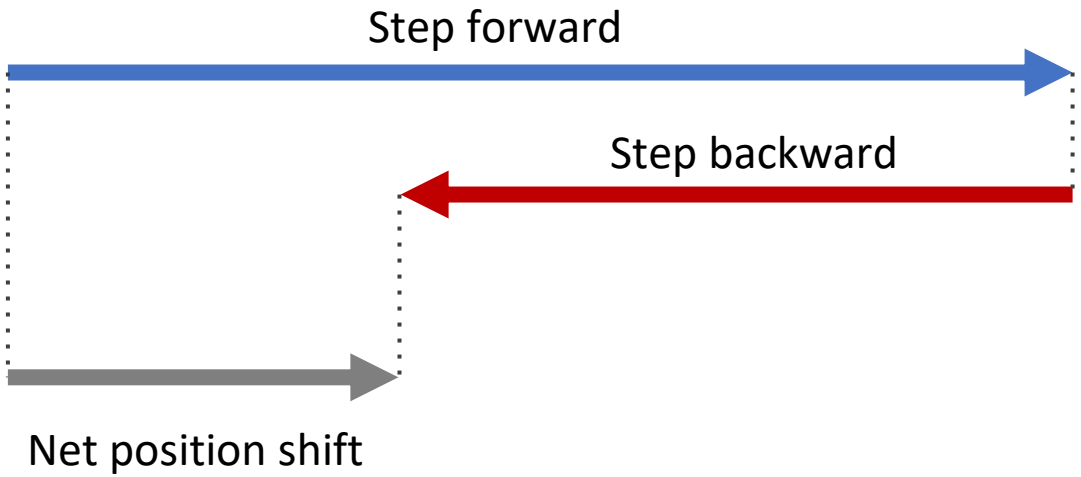


Shifting your body with SAMBA

- **SAMBA** is also a dance characterized by steps forward and backward

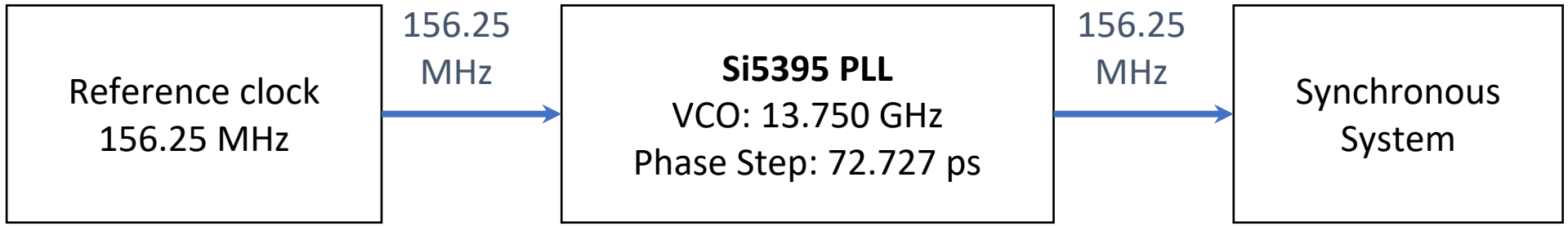
- If you wa our forward steps to be lon

DISCLAIMER:
The presenter does NOT actually know how to samba.



SAMBA Clock Shifting Method in a Nutshell

- Typical PLL-based clock distribution



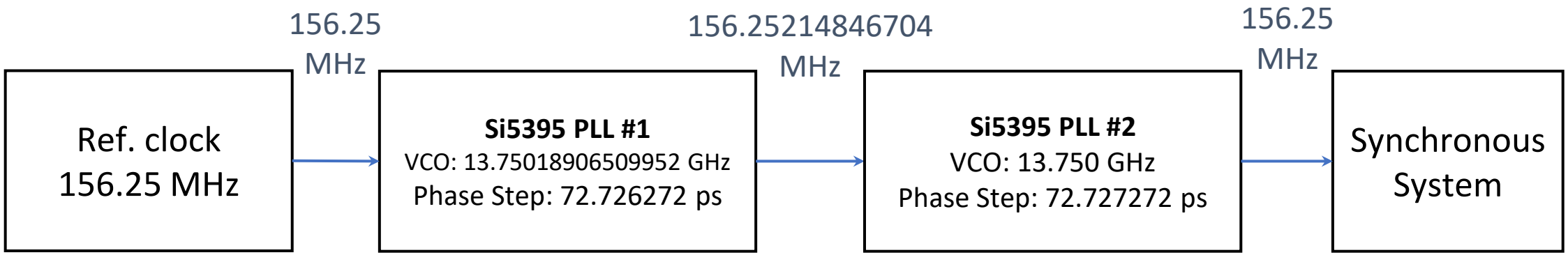
The Si5395 shifts the phase in steps of the VCO period.

For a 156.25 MHz output, the VCO is 13.75 GHz, therefore…

Phase shift resolution: **72.727 ps**

SAMBA Clock Shifting Method in a Nutshell

- Improved clock-shifting resolution with SAMBA



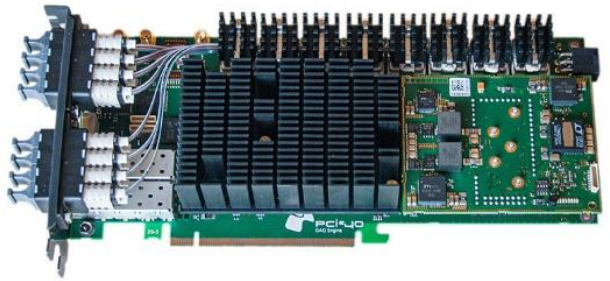
Shifting PLL #2 forward and PLL #1 backward gives a net shift of:

72.727272 ps
— 72.726272 ps
0.001000 ps

Phase shift resolution: **1 fs**

But just like the DDMTD phase measurement, SAMBA phase-shift is limited by signal integrity.

SAMBA on LHCb's PCIe40



SAMBA Shifting Resolution - Intel Arria 10

System **A**gnostic **M**ethod for **B**iphasic **A**lignment

First SAMBA Version

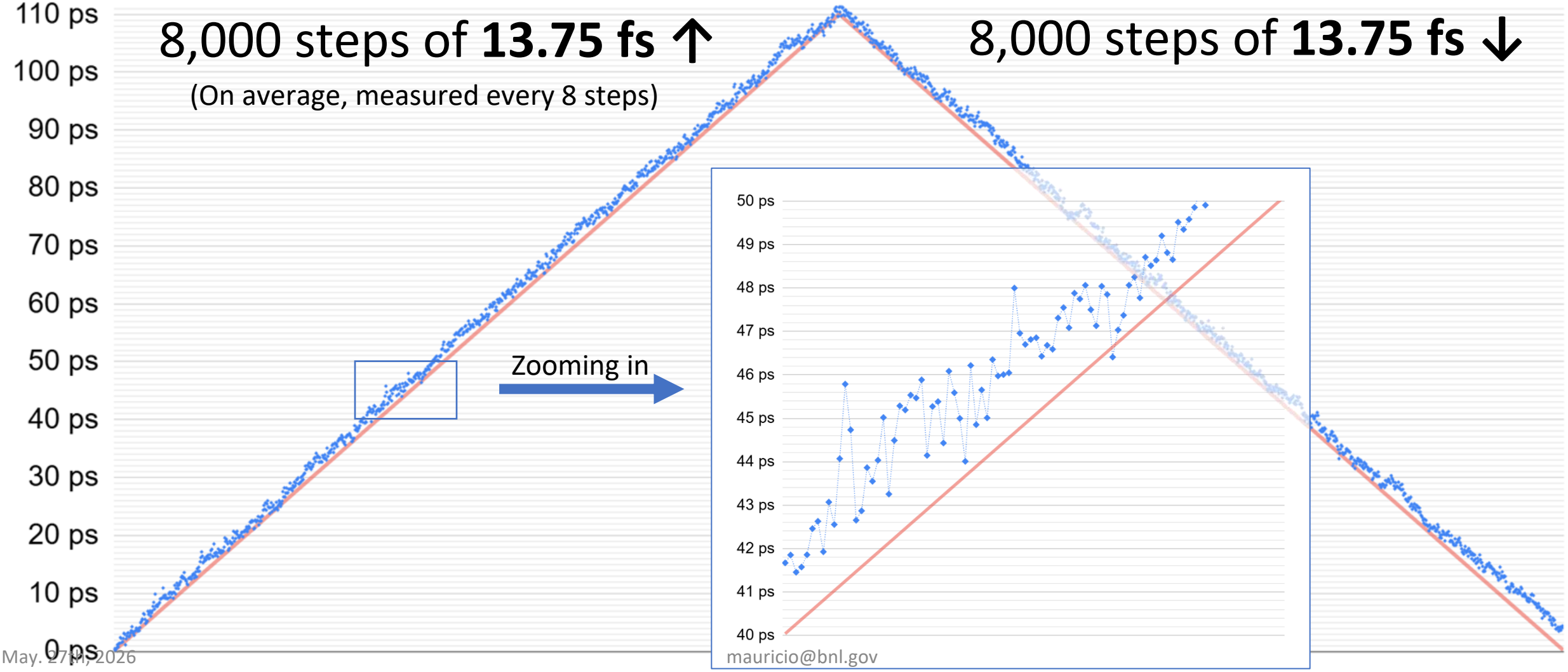
Increasing the Arria 10 phase-shift resolution by 2 orders of magnitude

Arria 10 native shift resolution: ~100ps

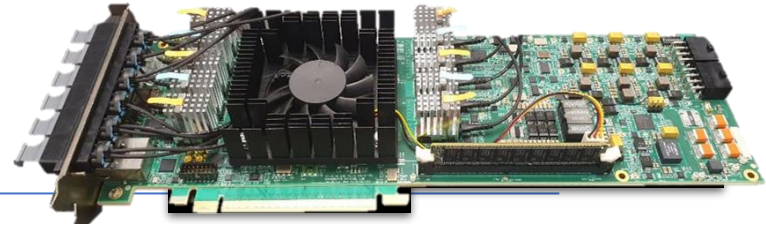
SAMBA shift resolution: ~1.5ps on average

70 steps of **1.5 ps** ↑

SAMBA on Intel's Cyclone 10 GX



SAMBA on ATLAS' FLX-155



◆ SAMBA Output Phase – Ideal Linear Phase

100 ps
90 ps
80 ps
70 ps
60 ps
50 ps
40 ps
30 ps
20 ps
10 ps
0 ps

100,000 steps of **1 fs** ↑

(On average, measured every 100 steps)

100,000 steps of **1 fs** ↓

Zooming in

50 ps
49 ps
48 ps
47 ps
46 ps
45 ps
44 ps
43 ps
42 ps
41 ps
40 ps

Signal Integrity

Ref. Clock Jitter: 300fs

Output Jitter: 100fs

SAMBA Phase Stability

- **Phase after a RESET:**
  - The phase stability after a reset depends on the PLLs used
  - SAMBA does NOT rely on post-reset phase stability
    - The phase needs to be precisely adjusted after each reset

- **Phase over time:**
  - If both PLL output frequencies don’t share a common divider, phase will drift over time.
  - The drift can be calculated to be insignificant or easily corrected.

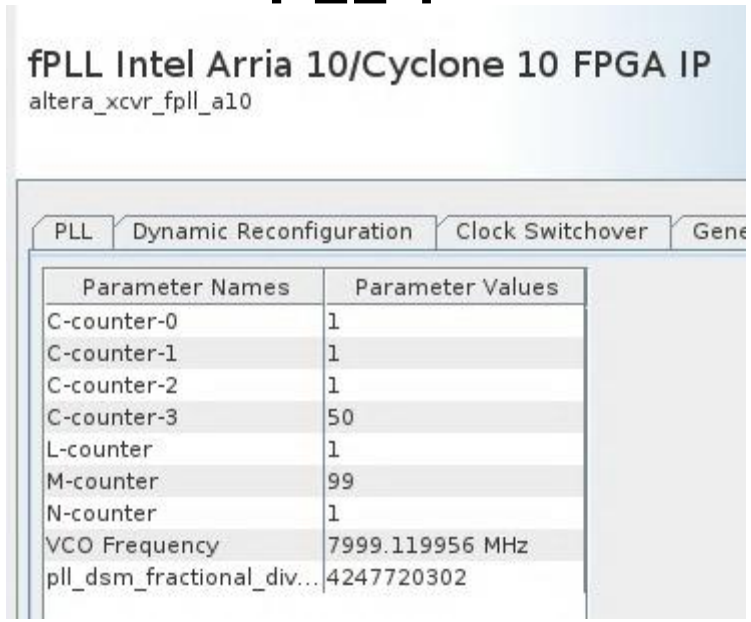
Choosing SAMBA Parameters

The challenge is finding values for the desired shift step while keeping the drift low enough

- **PLL Output Freq** $f_{out} = \left(\frac{M + F}{D}\right) \cdot f_{in}$
- **SAMBA Output Freq** $f_{out} = \left(\frac{M_1 + F_1}{D_1}\right) \cdot \left(\frac{M_2 + F_2}{D_2}\right) \cdot f_{in}$
- **Phase Shift Step** $= \frac{1}{VCO_1} - \frac{1}{VCO_2} = \frac{D_1 - (M_1 + F_1) \cdot f_{in}}{(M_2 + F_2) \cdot (M_1 + F_1) \cdot f_{in}}$
- **Phase Drift** $= \frac{f_{out} - f_{in}}{f_{in}} = \left(\frac{M_1 + F_1}{D_1}\right) \cdot \left(\frac{M_2 + F_2}{D_2}\right) - 1$

Choosing SAMBA Parameters (Cyclone 10 GX)

PLL 1

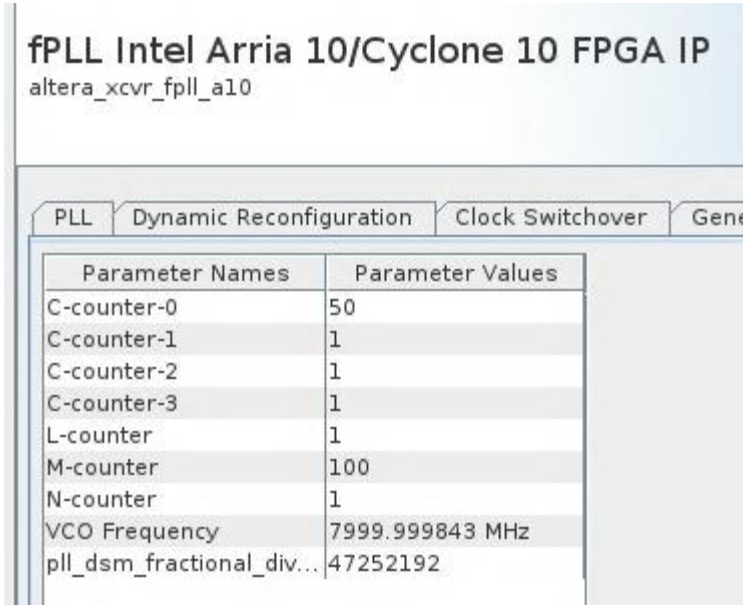


40 MHz →

39.995 MHz →

39995599.78079051 Hz

PLL 2



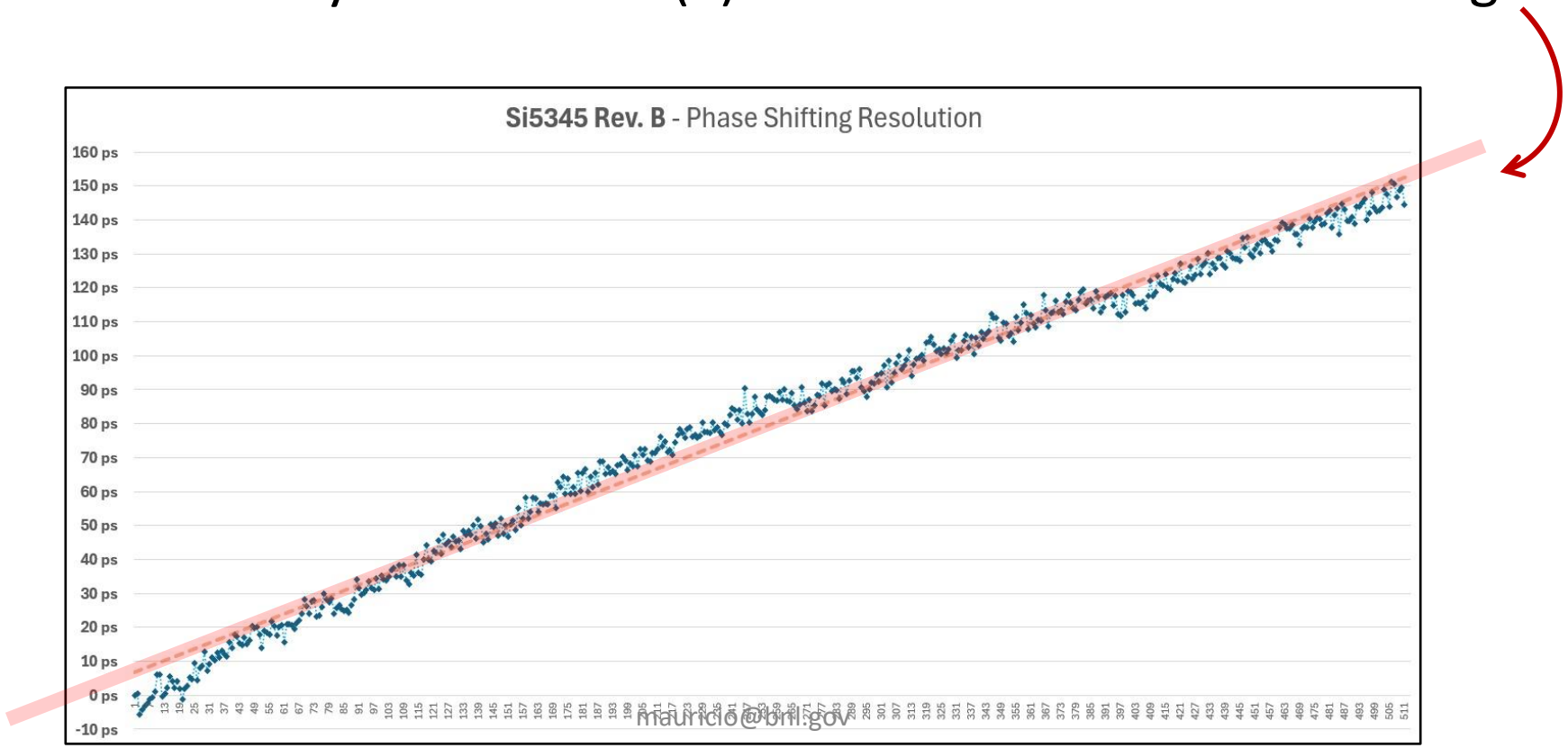
→ ~40 MHz

M1	M2	C1	C2	K1	K2	Resolution	Drift
99	100	50	50	4247720302	47252192	13.75 fs	-612 fs / hour
100	99	50	50	2152470774	2153230077	-623 fs	74 fs / hour
126	124	62	63	-	-	1.5 ps	0 fs / hour

*** 64 bits are not enough! I used the "decimal" Python lib for these calculations.**

SAMBA Linearity

- **Phase step linearity also depends on the PLLs used**
  - Si5395: Skips VCO periods to shift (?) -> Perfectly linear
  - Si5345: Uses delay-line to shift (?) -> Not linear and limited range



mauricio@bnl.gov

SAMBA Linearity

- **Intel FPGA PLLs:**
  - **IOPLL:** Linear every **8 steps**
  - **fPLL:** Linear every 4 steps

The lack of linearity from the PLLs used propagates to the SAMBA shifts, producing a periodic pattern.

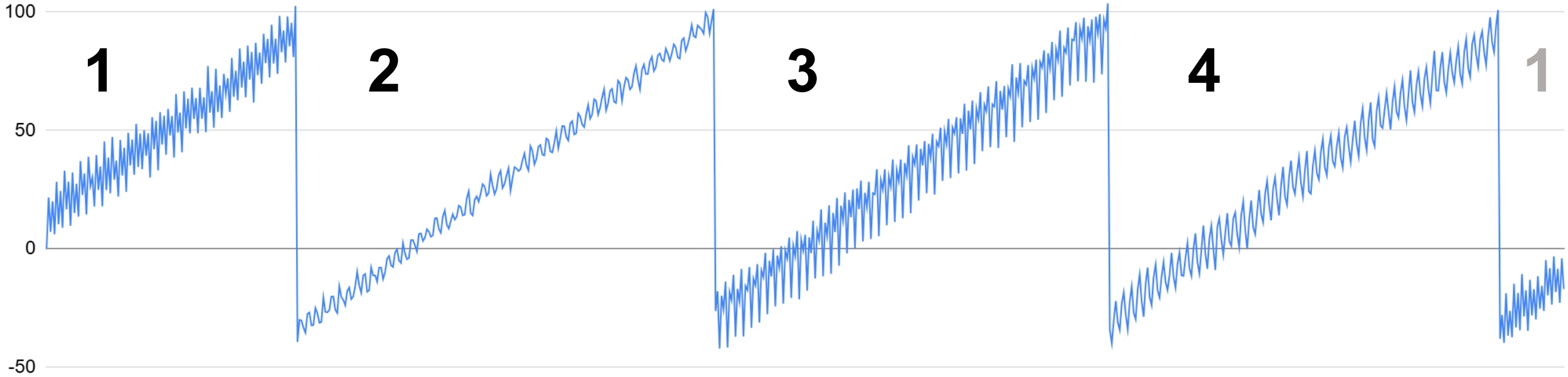
For linear shifts, either:
1. Shift in steps of the PLL number of taps (4 or 8) and sacrifice resolution
2. Make the shift resolution so low that the non-linearity is not perceivable

SAMBA Linearity

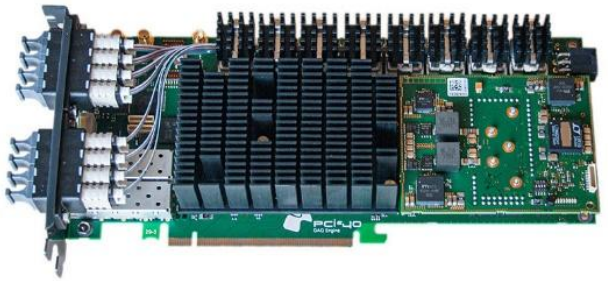
- **Intel FPGA PLLs:**
  - **IOPLL:** Linear every 8 steps
  - **fPLL:** Linear every **4 steps**

The lack of linearity from the PLLs used propagates to the SAMBA shifts, producing a periodic pattern.

For linear shifts, either:
1. Shift in steps of the PLL number of taps (4 or 8) and sacrifice resolution
2. Make the shift resolution so low that the non-linearity is not perceivable

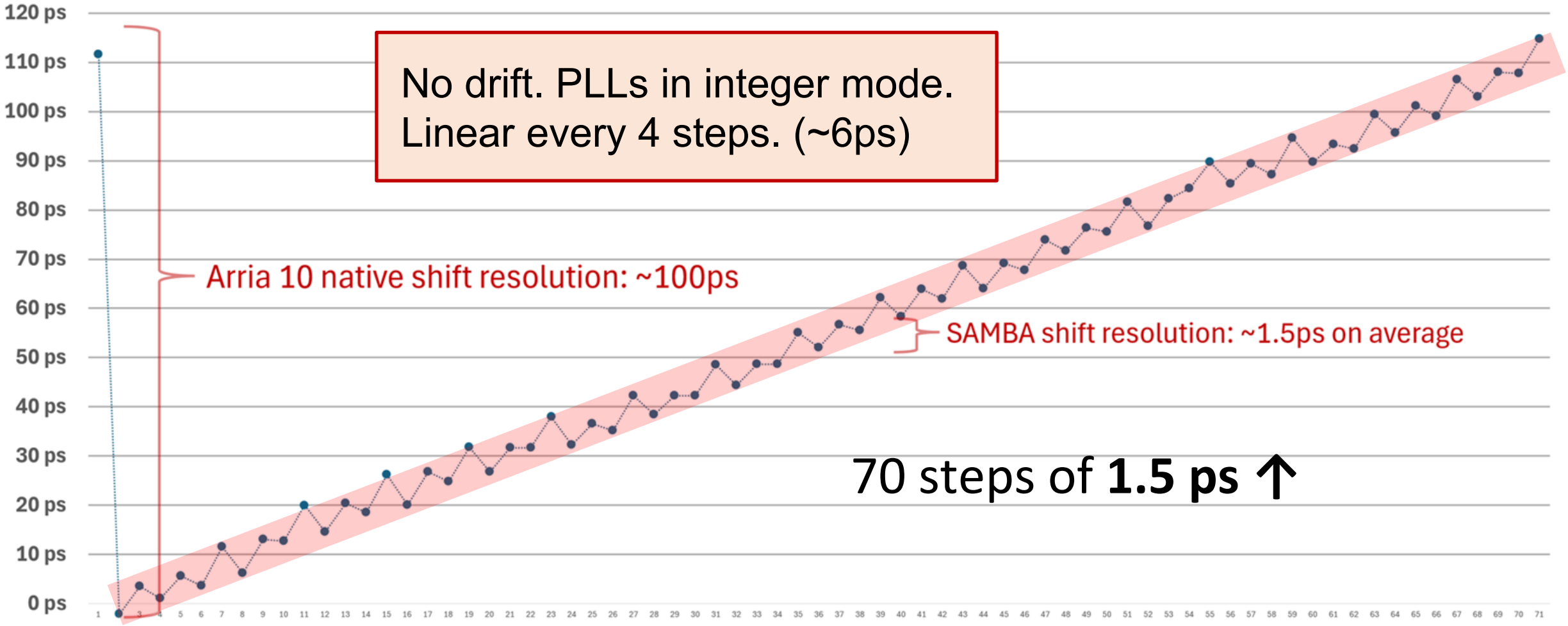


SAMBA on LHCb's PCIe40

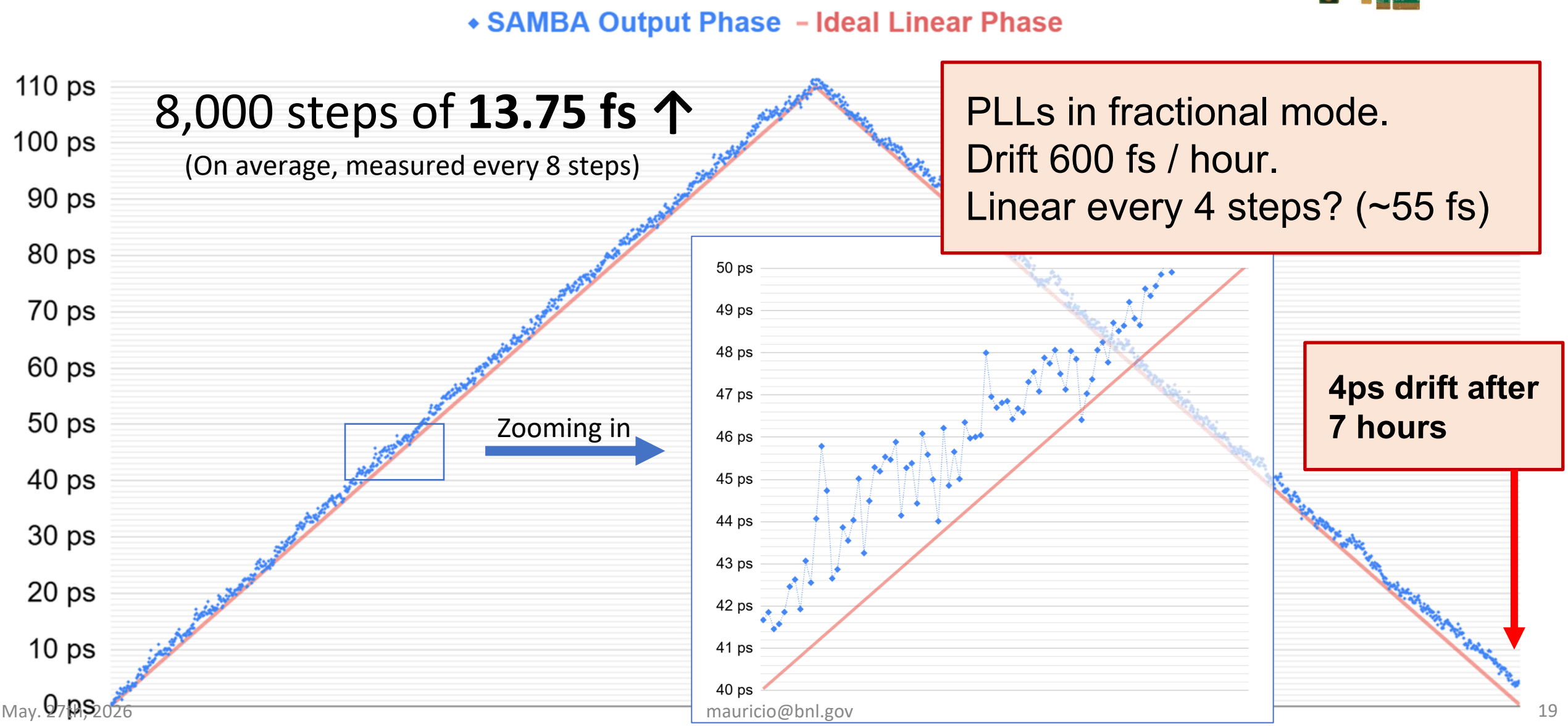


SAMBA Shifting Resolution - Intel Arria 10

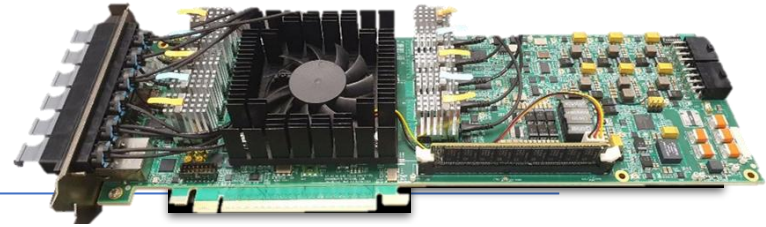
System **A**gnostic **M**ethod for **B**iphasic **A**lignment

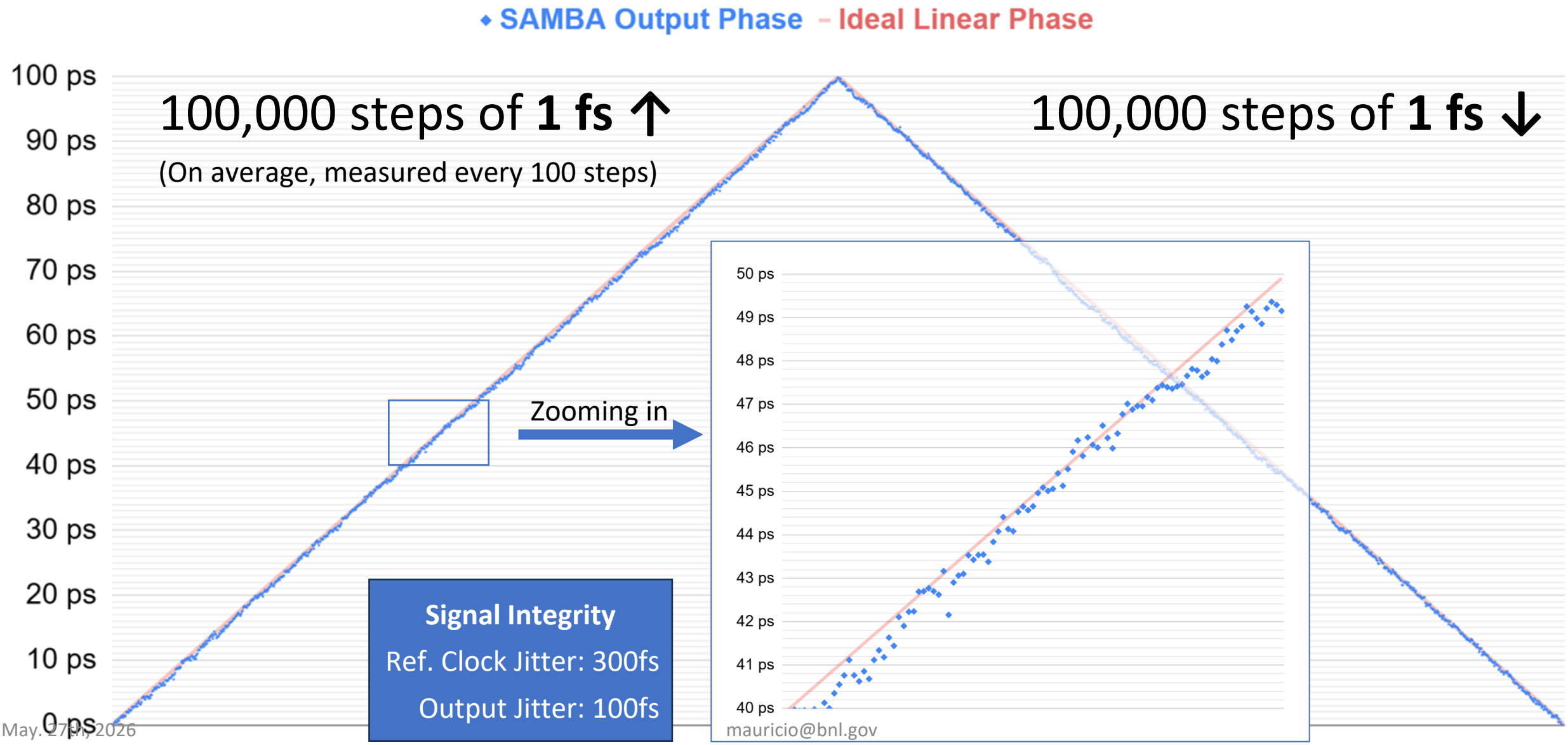


SAMBA on Intel's Cyclone 10 GX



SAMBA on ATLAS' FLX-155





mauricio@bnl.gov

Conclusion and SAMBA Future Prospects

- **SAMBA** is still in its early stages

- Further measurements with a climate chamber, better clock source, and in a low noise environment are planned in order to evaluate SAMBA's potential and limitations

- To be done:
  - Evaluation of the best PLLs for SAMBA
  - Production of a custom PCB with a SAMBA shifter
  - Characterization with precise measurements
  - Study the viability of an integrated implementation in silicon

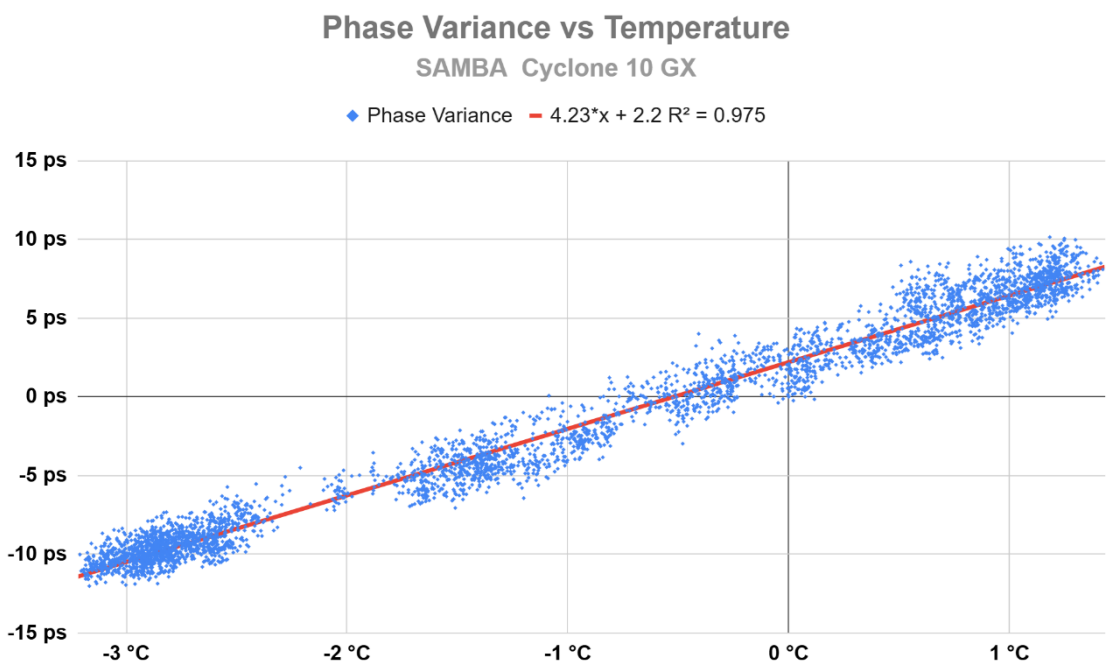
- **Your ideas, suggestions, and criticism are very welcome!**

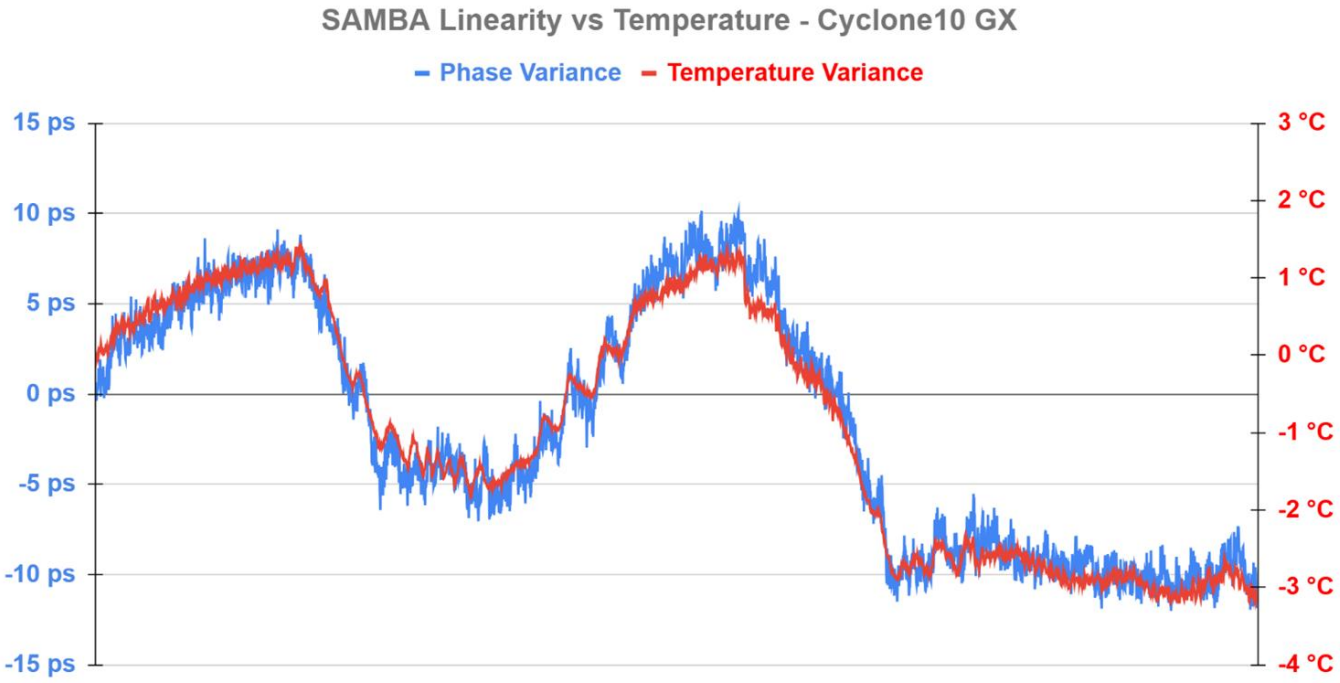
Obrigado!

Questions and comments are welcome

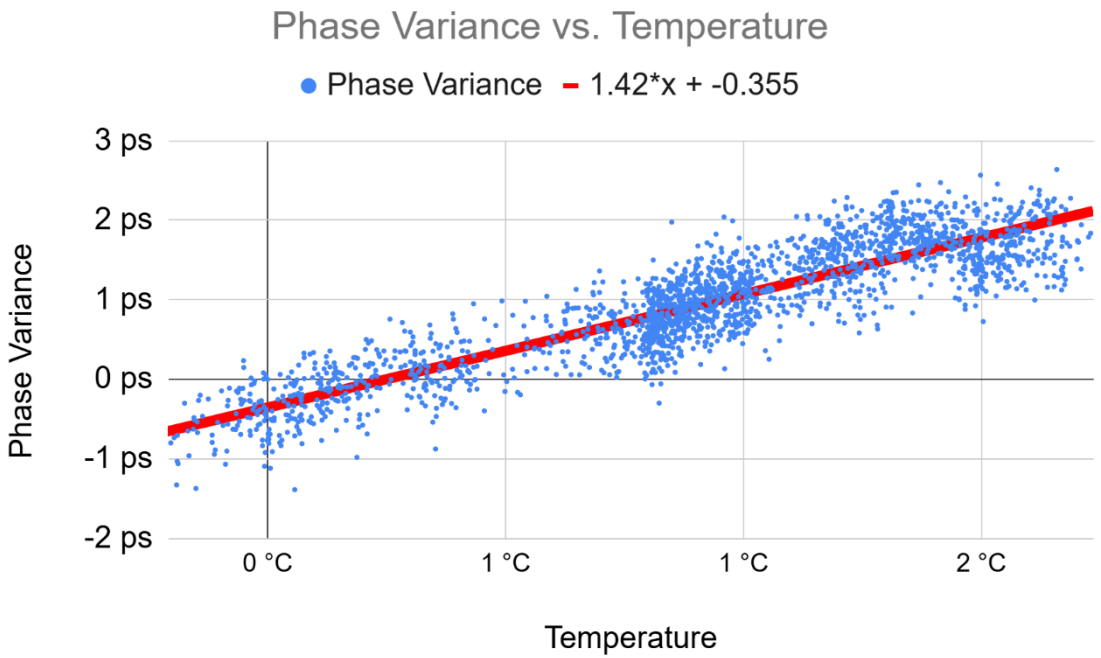
Backup

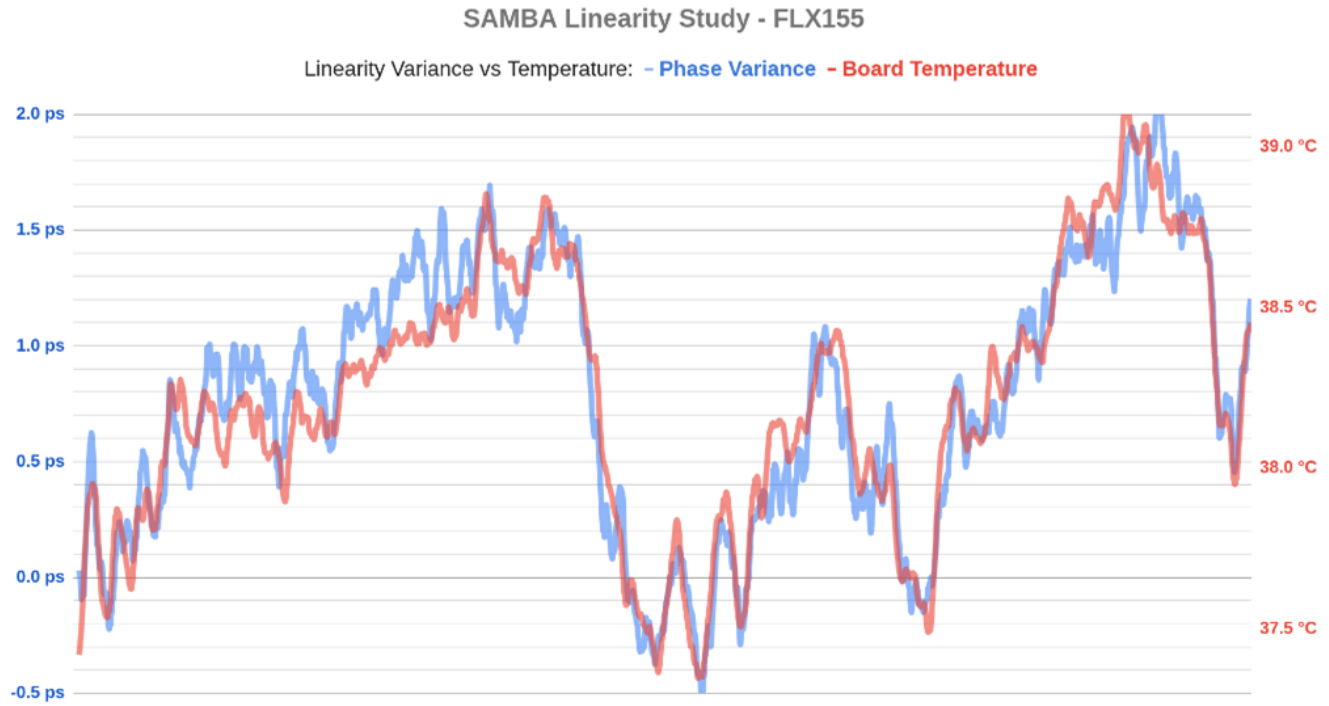
SAMBA Linearity vs Temperature – Cyclone 10





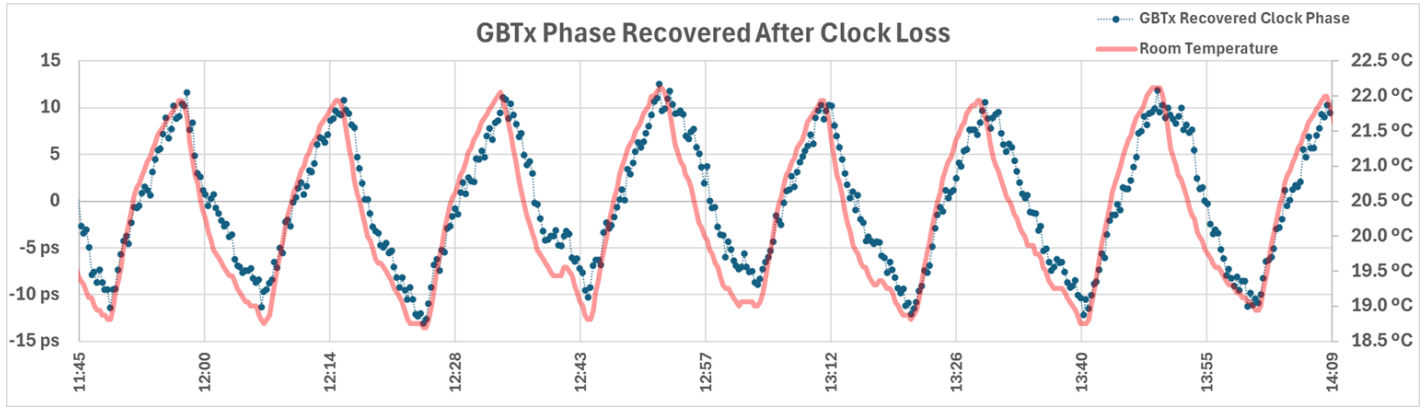
SAMBA Linearity vs Temperature - FLX-155



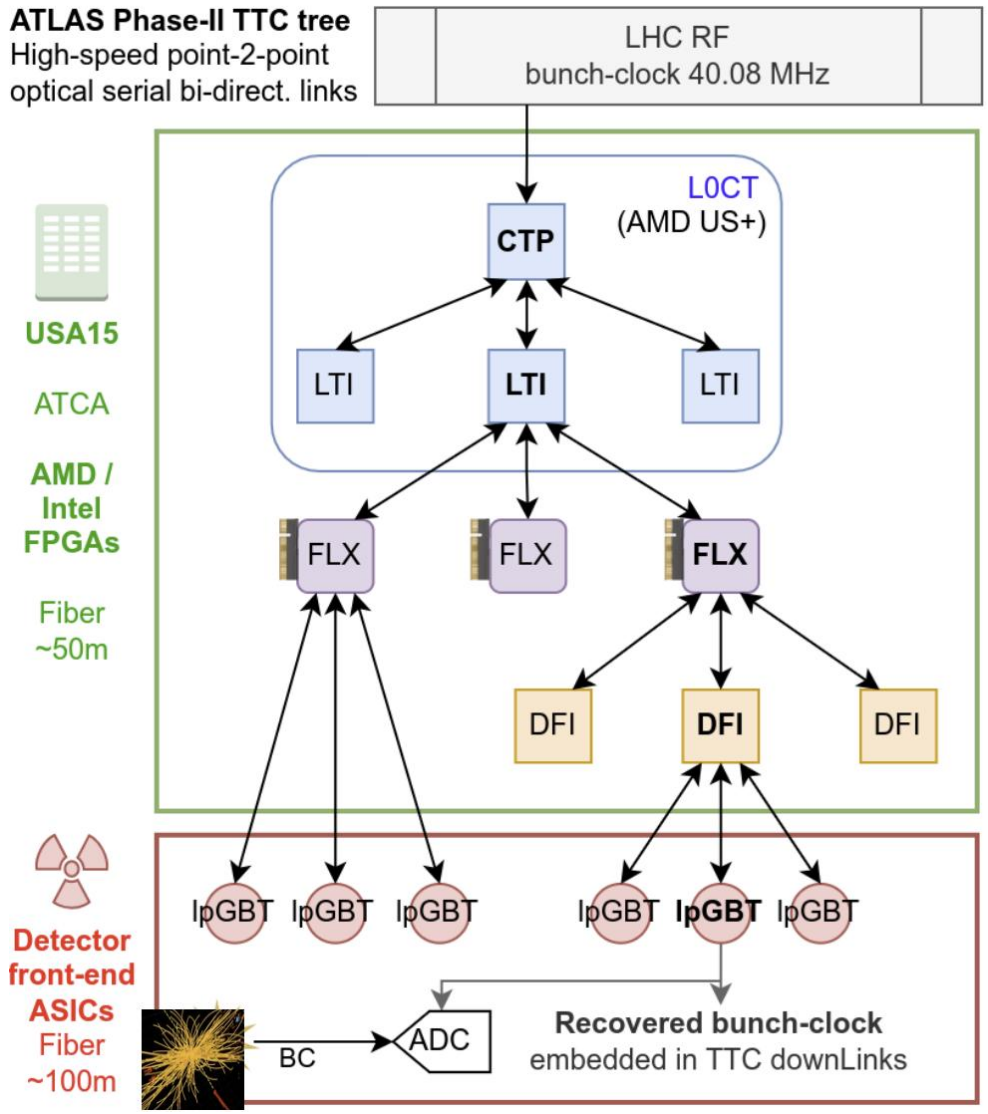


Timing Distribution at CERN Experiments

- The front-end clock needs to be phase-aligned with the LHC clock
- But phase drifts with temperature



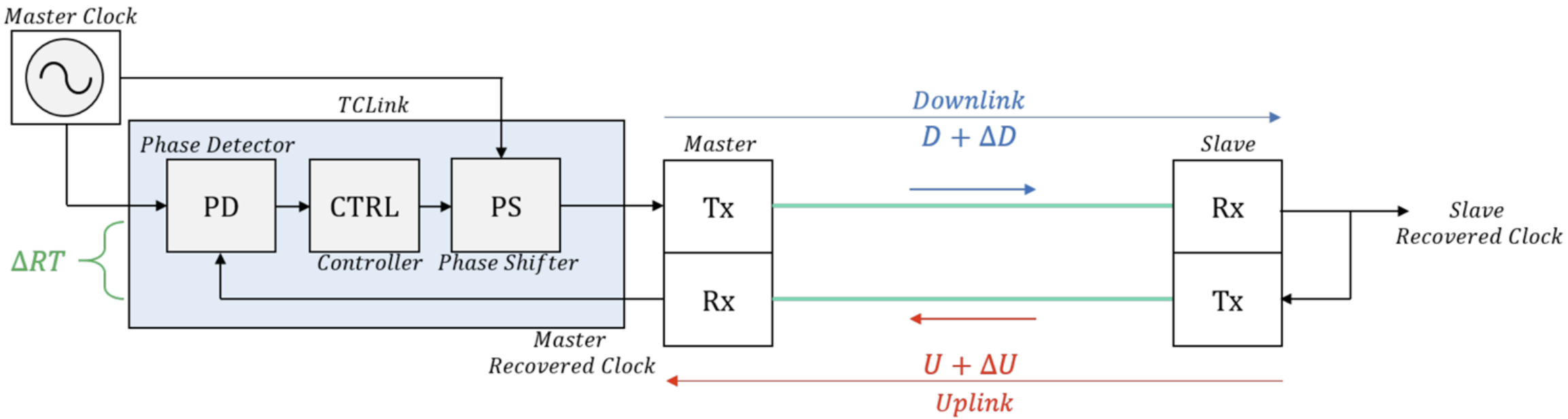
Front-end phase vs Room Temperature (from LHCb's PCIe40)



*From Filiberto's talk

Timing Distribution at CERN Experiments

- A timing system should measure the phase drift and shift the FE phase to compensate

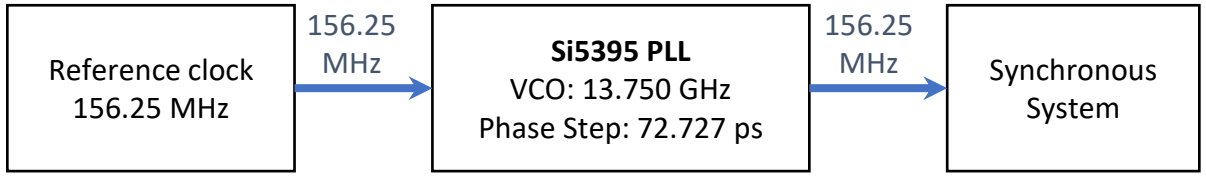


TCLink: a solution from CERN EP-ESE

*From Eduardo Mendes' paper

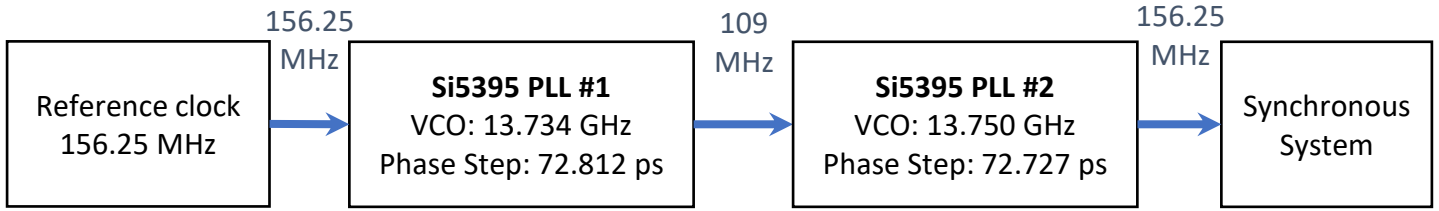
SAMBA Clock Shifting Method in a Nutshell

- Typical PLL-based clock distribution



Si5395 phase shift resolution: **72.727 ps**

- Improved clock-shifting resolution with SAMBA



Shifting PLL #1 forward and PLL #2 backward gives a net shift of 72.812 – 72.727 ps

SAMBA phase shift resolution: **0.085 ps**