Computational Locking: Accelerating Lock-times in All-Digital PLLs

Fahim ur Rahman[1], Greg F. Taylor[2], Visvesh S. Sathe[1]
[1] University of Washington, Seattle, USA [2] Intel Corporation, Hillsboro, USA
Email: fahimr@uw.edu

Abstract: We propose computational-lock (C-Lock), a technique for achieving rapid phase-acquisition in ADPLLs during cold-start and re-lock. A wide-dynamic range, high resolution TDC is also proposed to further support C-Lock. Lock-time (T_{lock}) performance is evaluated through 50,000 measurements of C-Lock enabled ADPLL test-chips in 65nm CMOS. Mean T_{lock} values of $16T_{refclk}$ and $12T_{refclk}$ for cold-start and re-lock respectively are reported. Used only during cold-start or re-lock, C-Lock does not impact steady-state PLL power and performance.

Keywords: Fast-lock; computational locking; PVT tolerant; wide-range TDC; high-resolution TDC

Introduction

Applications ranging from multi-core servers, mobile SoCs, and an increasing number of IoT applications can experience significant power and performance benefits from reducing PLL T_{lock} during wakeup (cold-start) and re-lock. Existing PLLs feature T_{lock} values of approximately 100 T_{refclk}[1]. Fast lock-techniques have been proposed [2-4] but they assume no temperature variation [2], require prior knowledge of PVT gain [3] or incur significant steady-state performance degradation [4].

In this paper, we propose computational lock for ADPLLs. In contrast with traditional type-II loop architectures, runtime computation of accurate phase-frequency PLL equations is employed to robustly achieve phase-lock 8x more rapidly. To further support C-Lock, we propose a novel wide dynamic-range, high resolution and fast resolving TDC architecture. C-Lock does not impact steady-state PLL operation and can be applied to a broad range of ADPLLs. Achieved T_{lock} values are independent of T_{refclk}. We demonstrate the proposed technique on a 1-2 GHz ADPLL intended for system clocking applications in 65nm CMOS.

Computational Lock (C-Lock) Architecture

C-Lock is implemented using an accelerator module (Solver) that augments the ADPLL (Fig.1(a)). At the onset of cold-start or a frequency change, a controller transfers loop control from the Digital Loop Filter (DLF) to the Solver (Fig 1(b)). After computationally determining DCO code sequences to achieve frequency, and subsequently phase lock, the controller seamlessly reverts to type-II loop control through the DLF, gating-off the Solver. Steady-state performance and power is not impacted by C-Lock.

Traditional PLLs feature higher mean and variance in T_{lock} largely due to (a) PVT induced loop gain variation (b) cycle-slipping and (c) the inherently non-linear behavior of ADPLLs. C-lock relies on accurate frequency-phase PLL equations and exploits computation to dynamically solve these equations, incorporating PVT dependent loop-gain variation, non-linear operation, and loop latency. Resulting T_{lock} distributions exhibit lower mean and variance.

The Solver begins lock acquisition by asserting an initial Digitally Controlled Oscillator (DCO) code-estimate, and performs a coarse phase-alignment of REFCLK and the divided DCO clock to within one time period of the DCO (T_{DCO}) through feedback counter modification (Fig. 2(a)), *Coarse align*). This step limits initial PLL frequency and phase, allowing low-complexity Solver calculations to provide sufficiently accurate lock solutions. Next, DCO frequency error Δf is calculated using the difference between successively sampled TDC codes (Equation 1-3, Fig. 3). The solver solves for DCO codes required to eliminate Δf. Since PLL gain cannot be predicted at runtime, a gradient-descent like approach is used to achieve lock (Fig. 4). Equations 1-4 (Fig.3) describe the accurate phase-frequency PLL equations used by the solver, which importantly depend on the convergence factor (μ_n), TDC and DCO gains (g_{TDC}, g_{DCO}) and the code-update latency (X) within the T_{refclk}. Phase-lock commences once the DCO frequency lies within a threshold of its target f_{lock} ($N*f_{REFCLK}$).

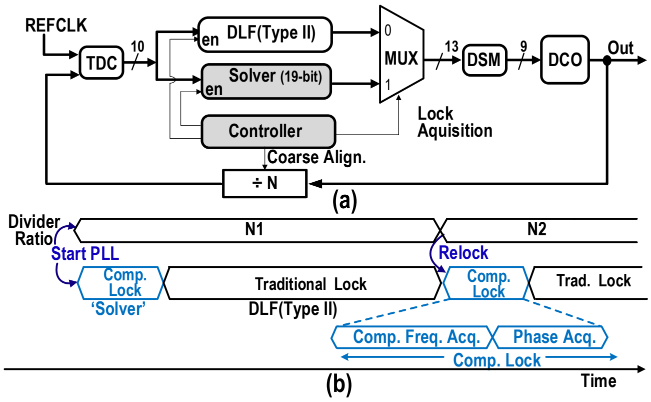


Fig. 1. (a) Block diagram and (b)operation of the proposed PLL.

Phase-lock involves using Eqn 5-6 (Fig. 3) to calculate required phase adjustments, performed by controlled frequency adjustments from f_{lock} for a portion of REFCLK (Fig. 2b). Once locked to within 0.002UI, the PLL *seamlessly* transitions over to traditional type-II operation.

Further support for C-Lock is provided by the proposed TDC offering a wide-input range (8ns), sub-gate delay resolution (1/3F04) TDC and fast resolution time (2ns, after budgeting for metastability resolution). The RO-Vernier based TDC (Fig. 5) consists of a ring-oscillator driving short Vernier delay-chains. Delay between input clock edges (*clkE*, *clkL*) is encoded into three successively finer delay units: (i)Coarse (RO-cycle count), (ii)Medium (Inverter-pair delay count) and (iii)Fine (Vernier delay resolution), offering 10 effective bits of resolution. The TDC offers wide dynamic range through a cycle counter (coarse), and requires short (7-stage) Vernier lines to cover the inverter-pair delay range. Incorrect latching of the asynchronous coarse-count is prevented by using the Medium code to judiciously select between two complementary-clocked Coarse-counters. The proposed Vernier-gater design (V-G) reduces power by triggering the Vernier lines only once, after the arrival of *clkL*.

Test-chip Implementation and Measurement

The ADPLL test-chip uses a 9-bit ganged-inverter based DCO [5] with 0.9-2.1GHz frequency range. Post-dividers enable continuous frequency coverage below 0.9GHz. 13-bit Solver/DLF values are used by a Digital Delta Sigma Modulator (DSM) to generate dithered patterns for improved DCO frequency resolution. A place-and-route Solver implementation incurs 25% power overhead during lock, with negligible impact on total ADPLL power given the relatively brief and infrequent occurrence of re-lock.

To enable a robust CPLL performance evaluation, we performed 50,000 iterations of PLL cold-start and re-lock across 15 test-chips with ±5% Vdd, and 0C-90C (30C increments) to incorporate PVT variation. Each cold-start frequency target and re-lock frequency combination was exercised equally. An on-chip BIST module was developed to enable high accuracy runtime measurement of PLL phase-error, T_{lock} and TDC and DCO transfer functions.

Measured relock-time distributions under nominal conditions exhibit mean (worst-case) T_{lock} of 12 (22) T_{refclk}, with PVT variation resulting in minor degradation to 12 (26) T_{refclk} (Fig. 6). Furthermore, repeated experiments at every possible from-to re-lock combination (Fig. 7) demonstrate mean T_{lock} to be largely independent of any specific frequency transition.

Importantly, C-Lock enables mean (worst-case) cold-start T_{lock} of 16 (27) T_{refclk} under nominal conditions and 16 (35) T_{refclk} under PVT variation (Fig. 8). Analysis from repeated lock events at varied T_{refclk} confirms analytical findings that T_{lock} is independent of T_{refclk} (Table I). A die photograph and comparison to related work are shown in Fig. 9 and Table II.

Acknowledgments: P. Howe, R. Pamula, T. Zhang, J. Koo, S. Pant, C. Tokunaga, S. Das, A. Loke, T. Karnik and V. De. This work was funded by an Intel gift award.

References

[1] K.J. Shen, ISSCC,2016, pp.330. [2] T. Anand, ISSCC, 2015, pp.64. [3]C-T. Wu, TCAS-II, 57, 430, 2003. [4] T. Watanabe, JSSC, 198, 2003. [5] J.A. Tierno, JSSC,43,42,2008. [6] F. Ahmad, ISSCC, 2016, pp.324. [7] J-M Lin, TCAS-I, 2415, 2015.

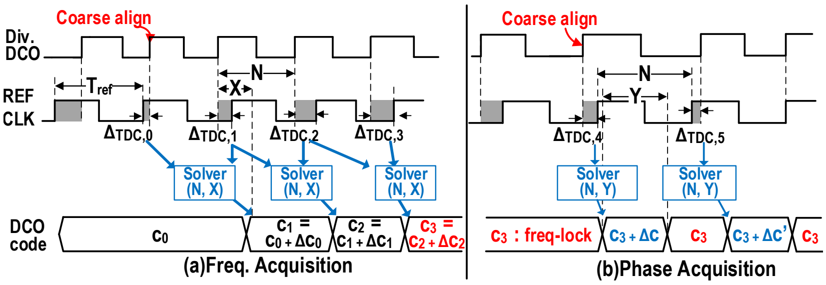
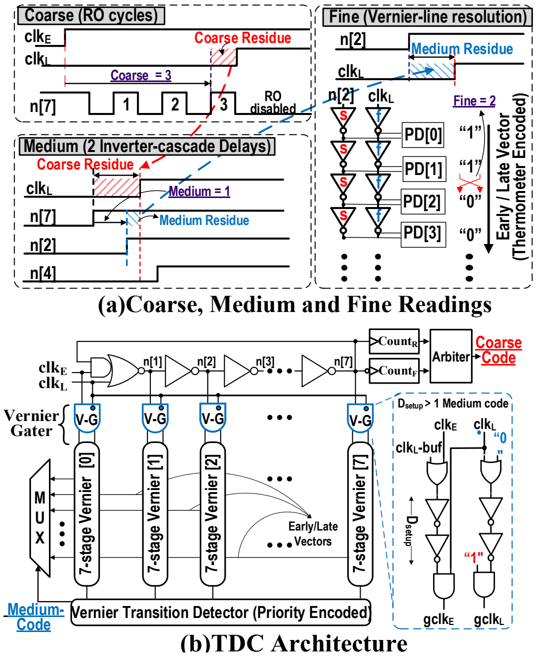


Fig 2. Chronological waveform of (a) frequency and (b) phase acquisition.



Definitions

N: PLL Divider Ratio,

X: REFCLK edge to DCO code change delay (T_{REF} units),

Y: Duration of phase adjustment (T_{REF} units)

Frequency-acquisition

$$T_{meas,n} = T_{REFCLK} + g_{TDC}(\Delta_{TDC,n} - \Delta_{TDC,n-1}) \quad (1)$$

$$NT_{DCO,n} = T_{meas,n} + \frac{X(T_{meas,n} - T_{DCO,n-1})}{N - X} \quad (2)$$

$$\Delta f_{DCO,n} \approx \frac{N(T_{REFCLK} - NT_{DCO,n})}{T^2_{REFCLK}} \quad (3)$$

$$\Delta C_{DCO,n} = \Delta f_{DCO,n} \cdot \mu_n \cdot g_{DCO} \quad (4)$$

Phase-acquisition

$$f_{bump,n} = f_{lock} + N(1 + \Delta_{TDC,n})/Y \quad (5)$$

$$\Delta C_{DCO,n} = f_{bump,n} \cdot \mu_n g_{DCO} \quad (6)$$

Fig 3. Derived system of equations for PLL phase/frequency update

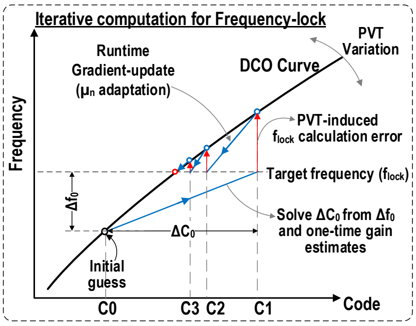


Fig 4. Graphical representation of frequency acquisition with gain adaptation.

Fig 5. Proposed RO-Vernier based TDC

F_{REF} (MHz)	Mean T_{lock} (REFCLK cycles)
25	13.3
30	11.8
35	11.6
40	12.8
45	12.8
50	12

TABLE I. F_{REFCLK} vs. mean T_{lock} for re-lock.

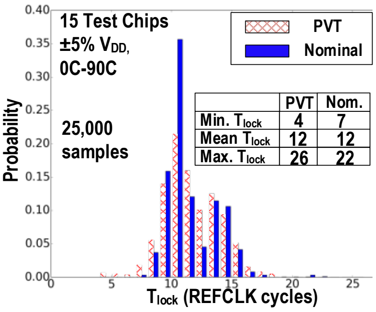


Fig 6. Histogram of T_{lock} for re-lock from 25,000 iterations.

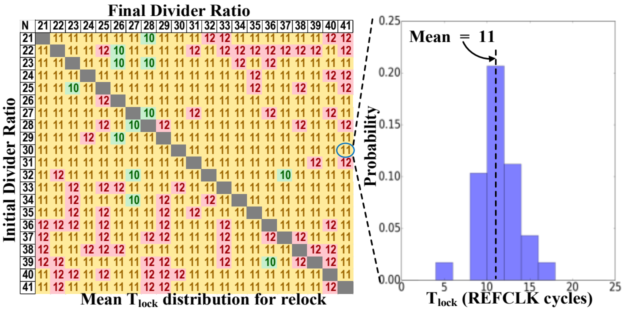


Fig 7. Mean T_{lock} for all possible frequency transitions. Each entry is obtained from multiple measurements.

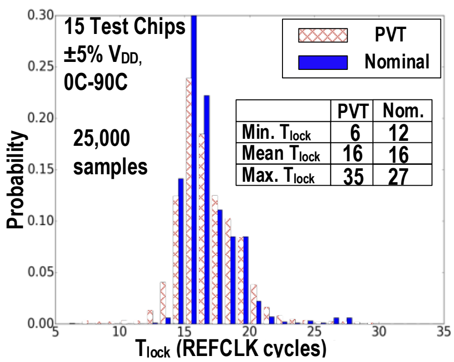


Fig 8. Histogram of T_{lock} for cold-start from 25,000 iterations.

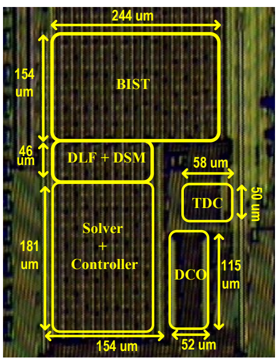


Fig 9. Die photograph.

	TCAS-II'10[3]	ISSCC'16[1]	ISSCC'16[6]	TCAS-I'15[7]	This work
Process Technology	0.18um CMOS	14nm CMOS	16nm CMOS	0.18um CMOS	65nm CMOS
PLL Architecture	ADPLL	Analog PLL	DPLL	ADPLL	ADPLL
Output Freq.(F_{REF})	28-446MHz (220kHz-8MHz)	0.15-5GHz (19.2MHz,100MHz)	0.5-9.5GHz (60MHz)	0.2539-1.367GHz (19.53MHz)	1-2GHz (50MHz)
Jitter(ps)	70(peak) @446MHz	3.765(rms) @1.6GHz	0.45(rms)	8.88(rms) 32.5(peak) @1.25GHz	3.09 (rms) 7.55 (peak) @1.5GHz
Best-case T_{lock}	2	NR	NR	NR	4(relock) 6(cold-start)
Mean T_{lock}	NR	100	75	57	12 (relock) 16(cold-start)
Worst-case T_{lock}	NR	NR	NR	NR	27(relock) 35(cold-start)
PVT Tolerant	NO	YES	YES	YES	YES
Power	14.5mW @446MHz	0.65mW @1.6GHz	7.1mW	35mW @1.25GHz	10.8mW @1.5GHz

TABLE II. Comparison table.