Verifying Global Convergence of a Digital Phase-Locked Loop with Z3

Yan Peng

University of British Columbia

September 27, 2013

Joint work with Jijie Wei, Grace Yu, and Mark Greenstreet

This work is funded by the Intel Corporation.

Outline

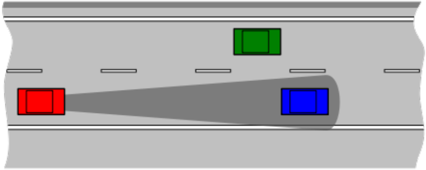
- Phase-Locked Loop (PLL) Introduction
- Verifying an All-Digital PLL with Z3[DMB08]
  - General approach for circuit verification with SMT solver
  - Global convergence verification for a simplified model with Lyapunov function
  - Limit cycle verification for a discrete recurrence model
- Ideas and Discussions
  - Comparison
  - Observations
  - Reachability Approach: SpaceEx
  - Conclusions

PLL: Thermostat Temperature Control

- First order control system
- Suppose T stands for current temperature, T_0 stands for desired temperature, then the system can be modelled as

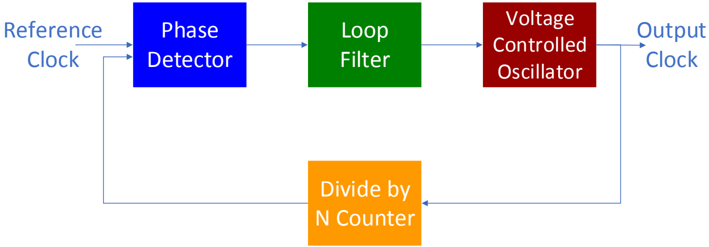
$$\dot{T} = -k(T - T_0).$$

PLL: Autonomous Cruise Control with Following



- Second order control system
- Suppose k_1 and k_2 are both positive, the system can be modelled as $k_2\ddot{S} + k_1\dot{S} + S = S_0$.
- This system has an equilibrium at $S = S_0$, and the two roots of $k_2\,\lambda^2 + k_1\lambda + 1$ both have negative real parts which make the system stable.

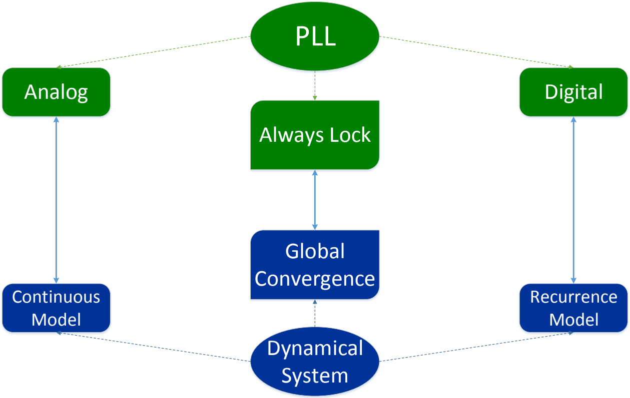
PLL: Block Diagram



A PLL is a feedback control system that, given an input reference clock, it outputs a clock at a frequency that's N times of the input clock frequency.

PLLs are ubiquitous in analog and mixed-signal designs. E.g. frequency multiplication for clock-acquisition in high-speed links, modulators and demodulators for wireless communication.

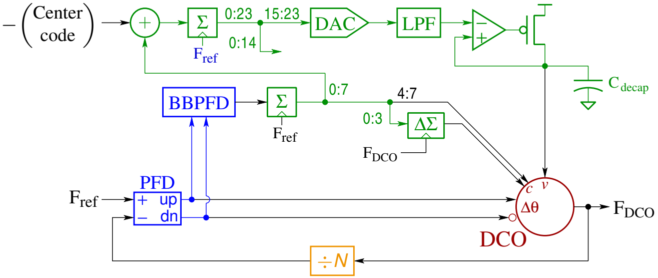
PLL: Global Convergence



The most essential function of a PLL is to lock at the right phase and frequency. The quicker a PLL locks the better.

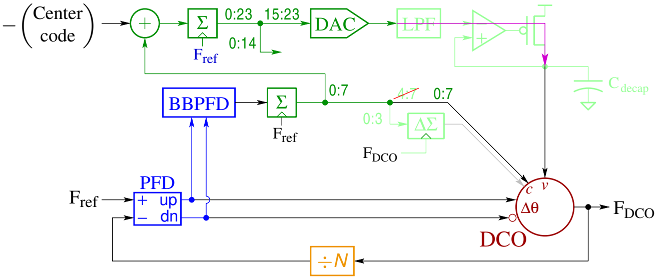
The global convergence problem asks whether a PLL locks no matter where its initial state is at.

PLL: Crossley et al.'s [CNA10]



- DCO has three control inputs:
  capacitance setting (digital), supply voltage (linear), phase correction (time-difference of digital edges).
- Uses linear and bang-bang PFD.
- Integrators are digital.
- LPF and decap to improve power-supply rejection.

PLL: Verified digital PLL

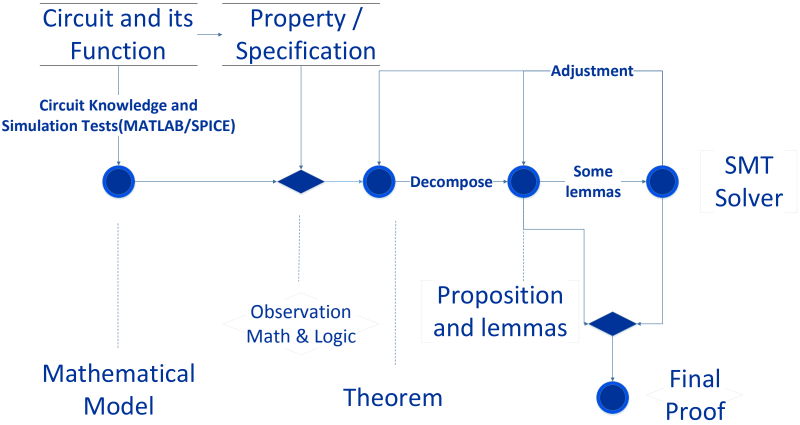


- We omitted the Delta-Sigma modulator, low-pass filter, and linear regulator for simplicity.
- We believe all of these could be included using the same methods as we've used for the rest of the digital PLL.

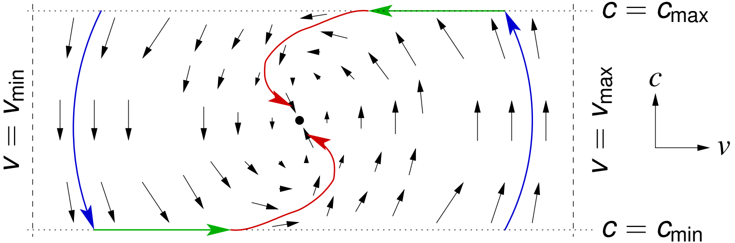
Outline

- Phase-Locked Loop (PLL) Overview
- Verifying an All-Digital PLL with Z3
    - General approach for circuit verification with SMT solver
    - Global convergence verification for a simplified model with Lyapunov function
    - Limit cycle verification for a discrete recurrence model
- Ideas and Discussions
    - Comparison
    - Observations
    - Reachability Approach: SpaceEx
    - Conclusions

Verification Approach



Dynamics of the Circuit



- Typical operation of the digital PLL

  C saturates. C is the "fast-tracking" control.

  v_{ctrl} moves toward equilibrium value.

  C and v_{ctrl} move through a sequence of limit cycles to maintain phase and frequency lock while centering C.
- Verification has separate "lemmas" for each of these phases.

Global Convergence Proof: Simplified Model

The ODE model:

$$\begin{aligned} \dot{c} &= g_1 \cdot \left(f_{ref} - \frac{1}{N}\frac{v_0+\alpha v}{c_0+\beta c}\right) \quad \textit{where} \quad c \in (c_{min}, c_{max}) \\ \dot{v} &= g_2 \cdot (c - c_{code}) \end{aligned}$$

Note: $\dot{c} = 0$ when $c = c_{min}$ or $c = c_{max}$

Linear phase path will be handled later, in the recurrence model analysis.

Quantization error will be included later.

Global Convergence Proof: Lyapunov Function

- Consider invariant region Q_0, target region Q_T with $Q_T \subseteq Q_0$. *Lyapunov function*, Ψ:
  - $\forall x \in Q_0 - Q_T.\ \Psi(x) > 0$
  - $\forall x \in Q_0 - Q_T.\ \frac{d}{dt}\Psi(x) < 0$
- For homogeneous linear ODE system $\dot{x} = Ax$:
  - Let P be the solution of $A^T P + PA = -I$. A possible Lyapunov function becomes $\Psi(x) = X^T PX$.
  - By construction, P is symmetric. If P is positive definite, then the system $\dot{x} = Ax$ globally converges to $x = 0$
- As long as one can find a Lyapunov function for a system, convergence to the desired equilibrium is ensured, no matter how the Lyapunov function is found.

Global Convergence Proof: The Full Proof

Translate the system

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \frac{d(c-equ_c)}{dt} \\ \frac{d(v-equ_v)}{dt} \end{bmatrix}$$

Derive the Jacobian matrix at operating point

$$A = Jac(f, X_0) = \left.\frac{\partial f}{\partial X}\right|_{X=X_0}$$

Calculate P matrix

Z3 solves this linear system readily

$$\boxed{\text{solve } A'P + PA = -Q, \text{ where } Q = I}$$

Form the statement to prove

Z3 checks if the two conditions are satisfied

$$\Psi(x) = X^T P X > 0 \quad when \quad X \neq 0$$
$$\dot{\Psi}(x) = 2\dot{X}^T P X < 0 \quad when \quad X \neq 0$$

Z3 easily proves global convergence when parameters are assigned fixed values and no quantization error is taken into consideration.

Global Convergence Proof: Improve the Model

- Z3 easily proves the simple system (just shown)
- We added details to the model to make it more realistic
  - Parameters with ranges
  - Quantization error: we approximate discrete sums in the real DPLL with integrals
  - Both ranges and quantization
- In all of these, Z3 would run longer than we had patience
- Solution: manually simplify terms in the proposed Lyapunov function
- The approach remains sound because Z3 checks the Lyapunov conditions, it doesn't matter how we came up with the Lyapunov function.

Global Convergence Proof: Adjust the Proof for Z3

Parameters with ranges. Inequality ranges for parameters α, β, v_0 and c_0.

Strategy: Check where the parameters are used, try simplify non-linear part. e.g. we replace some of the parameters in the Jacobian matrix with its nominal value.

Example:

Knowing $\alpha \approx 1$ and $\beta \approx 1$:

$$\mathbf{Jac}(\mathbf{f}, \mathbf{X_0}) = \begin{bmatrix} \frac{g_1 f_{ref} \beta}{c_0 + \beta c_{code}} & -\frac{g_1 \alpha}{c_0 + \beta c_{code}} \\ g_2 & 0 \end{bmatrix} \Rightarrow \mathbf{Jac}(\mathbf{f}, \mathbf{X_0}) = \begin{bmatrix} \frac{g_1 f_{ref}}{c_0 + \beta c_{code}} & -\frac{g_1}{c_0 + \beta c_{code}} \\ g_2 & 0 \end{bmatrix}$$

Z3 happily proved the conditions, again.

Global Convergence Proof: Adjust the Proof for Z3

Adding quantization error. Quantization error introduces new variables and inequalities.

Strategy: Noticing symmetry in the formula, transfer quantization error in the non-linear term to linear term.

Proposition:

Let *Err* denote quantization error and assume *Err* is symmetric around 0: if $\eta \in Err$ then $-\eta \in Err$ as well. We can easily show:

$$\begin{aligned}
&\forall x \in Q_0 - Q_T . \forall \eta \in Err . h(x+\eta)^T Px < 0 \Leftrightarrow \\
&\forall x \in (Q_0 - Q_T) \oplus Err \\
&\forall \eta \in Err . (x + \eta \in Q_0 - Q_T) \Rightarrow (h(x)^T P(x+\eta) < 0)
\end{aligned}$$

The Minkowski sum of two sets, $A \oplus B$, is the set of elements that can be obtained as the sum of an element from A and an element from B:

$$A \oplus B \quad = \quad \{z \mid \exists a \in A.\ \exists b \in B.\ z = a + b\}$$

Global Convergence Proof: Adjust the Proof for Z3

Combination. Combining quantization error and parameter ranges together.

Strategy: Further reduce the complexity of our problem by simplifying non-linear formulas.

Example:

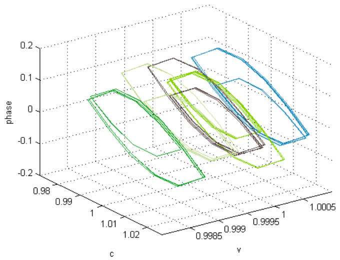
- Given $\beta c + c_0 + \beta c_{code} > 0$

$$(\beta c + c_0 + \beta c_{code})\dot{\phi}(X) < 0 \Rightarrow \dot{\phi}(X) < 0$$

- Given $c_{code} = 1$, $c_0 \approx 1$, and $\beta \approx 1$,

$$\mathbf{Jac(f, X_0)} = \begin{bmatrix} \frac{g_1 f_{ref}}{c_0 + \beta c_{code}} & -\frac{g_1}{c_0 + \beta c_{code}} \\ g_2 & 0 \end{bmatrix} \Rightarrow \mathbf{Jac(f, X_0)} = \begin{bmatrix} \frac{g_1 f_{ref}}{2} & -\frac{g_1}{2} \\ g_2 & 0 \end{bmatrix}$$

Limit Cycle Verification: the Recurrence Model



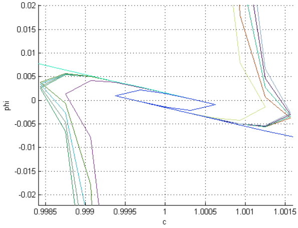
A limit cycle is an isolated closed trajectory, for which its neighbouring trajectories are not closed they spiral either towards or away from the limit cycle.

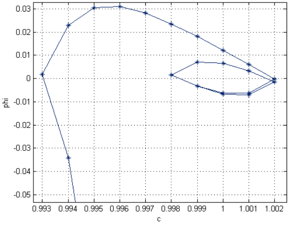
The recurrence model:

$$\begin{aligned} c(i+1) &= c(i) + g_1 sign(\phi(i)) \\ v(i+1) &= v(i) + g_2(c(i) - c_{code}) \\ \phi(i+1) &= (1 - K_t)\phi(i) + 2\pi\left(\frac{f_{dco}(i)}{Nf_{ref}} - 1\right) \end{aligned}$$

where $f_{dco}(i) = f_0 \frac{1+\alpha v(i)}{1+\beta c(i)}$

Limit Cycle Verification: Simplification with V

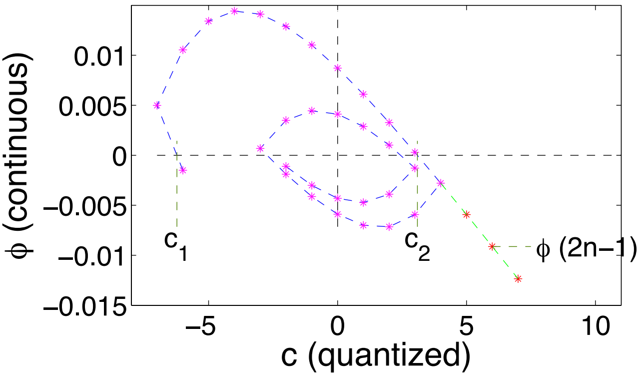




Simplified recurrence model:

v changes slowly with respect to c and ϕ. Thus, we can analyse the PLL as having a sequence of limit cycles, where each cycle is described by a recurrence of the form:

$$
\begin{aligned}
c(i+1) &= c(i) + g_1 sign(\phi(i)) \\
\phi(i+1) &= (1 - K_t)\phi(i) + 2\pi \left(\frac{f_{dco}(i)}{Nf_{ref}} - 1 \right)
\end{aligned}
$$



Theorem

If for each trajectory going through the upper space, suppose first point up crossing c-axis is $\phi(0)$, and we have $\phi(2n-1) < 0$, then the system must be converging towards some limit cycle.

Limit Cycle Verification: Induction Proof

Induction Proof

Suppose (c_0, ϕ_0) is the first point up crossing c-axis, and we have:

$$\begin{array}{rcl} c_0 & = & m \cdot g_1, \quad m \in \mathbb{Z} \ \wedge \ m \leq -3 \\ 0 & \leq & \phi_0 \quad < \quad 2\pi \left(\mu \dfrac{1 + \alpha v_0}{1 + \beta(m+1) g_1} - 1 \right) \end{array}$$

Solve the recurrence:

$$\begin{array}{rcl} c(j) & = & c_0 + g_1 j \\ \phi(j) & = & \gamma^j \phi_0 + 2\pi \displaystyle\sum_{i=0}^{j-1} \gamma^{(j-1-i)} \left(\mu \dfrac{1 + \alpha v_0}{1 + \beta c(i)} - 1 \right) \end{array}$$

We want to prove:

$$\forall m \geq 3, \ \phi(2m-1) < 0$$

A symmetric argument applies to lower half of the space.

Limit Cycle Verification: Lemma 1 (by induction)

We manually rewrote the inequalities and decomposed the proof into the formula below:

Z3 formula

$$\frac{\gamma^2\left(\frac{1+\alpha v_0}{1+\beta(m-1)g_1}-\frac{1+\alpha v_0}{1+\beta m g_1}\right)+\gamma\left(\frac{1+\alpha v_0}{1+\beta m g_1}-\frac{1+\alpha v_0}{1+\beta(m+1)g_1}\right)+\left(\frac{1+\alpha v_0}{1+\beta(-(n-1)g_1+equ_c)}-\frac{1}{\mu}\right)}{\frac{1}{\mu}-\frac{1+\alpha v_0}{1+\beta((n-1)g_1+equ_c)}}<\gamma^{2-2n}$$

Decompose this formula into two:

new Z3 formula

$$\frac{\gamma^2\left(\frac{1+\alpha v_0}{1+\beta(m-1)g_1}-\frac{1+\alpha v_0}{1+\beta m g_1}\right)+\gamma\left(\frac{1+\alpha v_0}{1+\beta m g_1}-\frac{1+\alpha v_0}{1+\beta(m+1)g_1}\right)+\left(\frac{1+\alpha v_0}{1+\beta(-(n-1)g_1+equ_c)}-\frac{1}{\mu}\right)}{\frac{1}{\mu}1-\frac{1+\alpha v_0}{1+\beta((n-1)g_1+equ_c)}}<2n$$

and

$$2n<\gamma^{2-2n}$$

Limit Cycle Verification: Lemma 2 (by induction)

An additional induction proof is provided for proving $2n < \gamma^{2-2n}$(where $0 < \gamma < 0.5$):

Additional Induction Proof

- When $k = 4$, $8 < \gamma^{-6}$
- Suppose when $k = n$, $2n < \gamma^{2-2n}$ stands, then we have when $k = n + 1$, $2(n+1) < \gamma^{2-2(n+1)}$.
  - Here is an simple illustration showing how Z3 helps discharge formula to be proved:
    $\because 2n < \gamma^{2-2n}$
    $\therefore 2n\gamma^{-2} < \gamma^{-2n}$
    $\because 2n + 2 \leq 2n\gamma^{-2} \Rightarrow$ This is the formula for Z3!
    $\therefore 2n + 2 < \gamma^{-2n}$

Limit Cycle Verification: Finding One Limit Cycle

We find one limit cycle by stating what properties should a cycle have:

Property of a limit cycle with 6 vertices

- The 6 points should satisfy the recurrence
- The next point for the 6th point should be the 1st point
- To make it easier, we specify small target ranges for the point positions
- We find in some limit cycles, there must be one point at which $c = 0$(suppose the system is translated to the origin)

Outline

- Phase-Locked Loop (PLL) Overview
- Verifying an All-Digital Phase-Locked Loop with Z3
    - General approach for circuit verification with SMT solver
    - Global convergence verification for a simplified model with Lyapunov function
    - Limit cycle verification for a discrete recurrence model
- Ideas and Discussions
    - Comparison
    - Observations
    - Reachability Approach: SpaceEx
    - Conclusions

Comparison

Lyapunov function method:

- Fixed procedure, easy for automation
- Take less human effort, much is done in Z3
- More suitable for continuous system without large non-linear part.

Induction proof method:

- More flexibility in procedure, not easy for automation
- Take lots of pencil and paper work to decompose the proof and takes the human mind to combine the proofs.
- Suitable for more sophisticated system, difficulty lies in how to find a theorem to prove the property.

Observations

- The part of Z3 we frequently used in our proof:
  - Real arithmetic
  - Linear and polynomial arithmetic
  - Propositional logic: *prove*, *Implies*
  - Very large propositional statement with polynomials and some inequalities
- Things that'll probably help with our work:
  - Real and Integer combined arithmetic
  - Non-linear arithmetic, E.g. exponentials, rational functions
  - Induction proof
  - Combining lemmas, like a theorem prover

Reachability Approach: SpaceEx

- SpaceEx[FLGD+11] is a hybrid-automaton based tool on reachability analysis for safety verification.
- Build hybrid-automata models for each component:
  - DCO: 7 modes – linearize for overlapping intervals of v_{ctrl}.
  - PFD: 1 mode, with self-loops.
  - C-accumulator: 4 modes – up, down, saturated low, saturated high.
  - V-accumulator: 3 modes – normal, saturated low, saturated high.
- Product machine has 84 modes.
- Dynamics of the system lead to transitions between modes.
- SpaceEx confirms convergence to lock from any initial state.

Reachability Approach: SpaceEx - Comparison

- The hybrid-automaton approach very closely followed the structure of the digital PLL.
  - Seems likely to be more intuitive for real-world designers and verifiers.
  - Verified the digital PLL for a specific choice of model parameters.
  - "Feels" like model-checking.
  - Manual decomposition into lemmas required.
- The SMT approach is more general:
  - Handles some of the non-linearities directly.
  - Verified the digital PLL with model parameters in interval ranges.
  - "Feels" like theorem proving.

Conclusion

- SMT solvers such as Z3 can be extensively adopted in the field of mixed signal circuit verification and continuous dynamic system verification.
- SMT approaches are promising for verifying properties of mixed signal designs, such as proving global convergence and identifying limit cycle behaviours. These properties are impossible to verify with traditional simulation.
- We would like to further automate the verification, especially tracking proof obligations for induction arguments and showing that SMT-discharged lemmas prove the main result.
- We would like to better understand the workings of the SMT solver to use it more effectively and possibly extend it to handle wider ranges of problems.

Questions and suggestions are welcome!

References

- J. Crossley, E. Naviasky, and E. Alon, *An energy-efficient ring-oscillator digital pll*, Custom Integrated Circuits Conference (CICC), 2010 IEEE, 2010, pp. 1–4.
- Leonardo De Moura and Nikolaj Bjørner, *Z3: an efficient smt solver*, Proceedings of the Theory and practice of software, 14th international conference on Tools and algorithms for the construction and analysis of systems (Berlin, Heidelberg), TACAS'08/ETAPS'08, Springer-Verlag, 2008, pp. 337–340.
- Goran Frehse, Colas Le Guernic, Alexandre Donzé, Scott Cotton, Rajarshi Ray, Olivier Lebeltel, Rodolfo Ripado, Antoine Girard, Thao Dang, and Oded Maler, *Spaceex: scalable verification of hybrid systems*, Proceedings of the 23rd international conference on Computer aided verification (Berlin, Heidelberg), CAV'11, Springer-Verlag, 2011, pp. 379–395.