A Compact, Low-Power Low-Jitter Digital PLL

Amr Fahim
Qualcomm, Inc.
Outline

• Introduction & Motivation
• Digital PLL Architectures
• Proposed DPLL Architecture
• Analysis of DPLL
• DPLL Adaptive Algorithm
• DPLL Circuit Implementation
• Measurement Results
• Conclusions
Introduction & Motivation
Introduction & Motivation

• Classical 2nd order analysis:

\[ H(s) = \frac{K R_s + K}{s^2 + \frac{K}{N} R_s + \frac{K}{NC}} \]

• where:

\[ K \equiv I_p K_{v_{2i}} K_{ico} \]

\[ \omega_n = \sqrt{\frac{K}{NC}} \]

\[ \zeta = \frac{1}{2} R \sqrt{\frac{K}{NC}} \]
Introduction & Motivation

• Sensitivity Analysis:

• Assumptions:
  – \( I_p = \pm 20\%, \; R = \pm 20\%, \; C = \pm 10\% \)

• Loop Dynamics Variation:
  – \( \zeta \in [-41\%, +59\%] \)
  – \( \omega_n \in [-48\%, +33\%] \)

• Effect on Lock Time:
  – Assumption: cycle slipping time is minimal
  – \( T_L \in [-25\%, +92\%] \)

• Effect on phase noise:
  – Mostly due to BW variation
  – Neglecting 1/f noise
  – Jitter variation is 25%
Introduction & Motivation

- Noise coupling to loop filter:
  
- Reduce effect of noise coupling on jitter by:
  - Reducing $K_v$
  - Increasing guard ring spacing

- Expected to worsen as processor complexity grows
Alternative Paradigm

- Digitize Loop Filter
  - Less analog circuitry – faster design time
  - Less susceptible to ground bounces
  - Better stability over process, temperature, and voltage
Outline

- Introduction & Motivation
- Digital PLL Architectures
- Proposed DPLL Architecture
- Analysis of DPLL
- DPLL Adaptive Algorithm
- DPLL Circuit Implementation
- Measurement Results
- Conclusions
Digital PLL Architecture

• Can classify DPLL architecture by method by which phase & frequency error is determined
  – Method A: Direct phase error digitization
  – Method B: Frequency comparison & Reset
Digital PLL Architecture

• **Method A**: Direct phase error quantization:

  - Count number of VCO cycles per reference cycle
  - Max phase error is:
    \[ \Delta \theta, \text{max} = \frac{T_{REF}}{N-1} - \frac{T_{REF}}{N} \]

  - Inherent trade-off between quantization jitter and ICO jitter

![Digital PLL Architecture Diagram]
Digital PLL Architecture

- **Method B**: Frequency compare & reset

  - Avoid quantization jitter bottleneck by using frequency locked loop and reset VCO every compare period
  - Quantization jitter reduced to how well can estimate frequency and how fast can reset VCO
  - Requiring VCO to be reset can limit upper frequency and VCO topology
Digital PLL Architecture

- **Method B**: Frequency compare & reset
Outline

- Introduction & Motivation
- Digital PLL Architectures
- Proposed DPLL Architecture
- Analysis of DPLL
- DPLL Adaptive Algorithm
- DPLL Circuit Implementation
- Measurement Results
- Conclusions
Proposed DPLL Architecture

- Reduce PFD to 1-bit output, which is weighted dynamically
- Small cycle-to-cycle phase error when locked, therefore DPLL behaves as a linear system
- Filter is composed of accumulator (integrator term) and a multiplier (proportional term)
Proposed DPLL Architecture

- Fast binary phase locking algorithm
- Reduce weight by half for every phase error sign change
Proposed DPLL Architecture

- **PFD Diagram:**

- **Asynchronous Timing mechanism:**
  - PFD Generates a “clock” token which is used to drive the digital filter
  - When filter value is updated a completion signal is sent back to PFD
Outline

- Introduction & Motivation
- Digital PLL Architectures
- Proposed DPLL Architecture
- Analysis of DPLL
- DPLL Adaptive Algorithm
- DPLL Circuit Implementation
- Measurement Results
- Conclusions
DPLL Linear Analysis

- Assume cycle-to-cycle phase error is small
- Discrete-time model:

\[
\frac{1}{N} F(z) ZOH K_i DAC K_o I K R K_2 Z N K_2 K R \]

- Closed loop expression:

\[
H(z) = \frac{K \cdot (K_R + K_I) \cdot \left[ Z - \frac{K_R}{K_R + K_I} \right]}{z^2 - \left( 2 - \frac{K \cdot (K_R + K_I)}{N} \right) \cdot Z + \left( 1 - \frac{K}{N} K_R \right)}
\]
DPLL Linear Analysis

- Case of $K_I = K_R = 1$
  - System is unconditionally stable
  - As open loop dc gain increases, poles migrate towards the origin
DPLL Linear Analysis

- $K_I=1, K_R \uparrow$

- As $K_R$ increases, radius of circle shrinks (i.e. less damping)

- Can be used to effectively control the damping factor of closed loop system

- $K_R$ has little affect on loop bandwidth
DPLL Linear Analysis

- $K_R=1$, $K_I \uparrow$

- $K_I$ affects both damping factor and loop bandwidth

- As $K_I$ is increased, radius of circle increases; i.e. poles move faster and more oscillatory
Outline

• Introduction & Motivation
• Digital PLL Architectures
• Proposed DPLL Architecture
• Analysis of DPLL
• DPLL Adaptive Algorithm
• DPLL Circuit Implementation
• Measurement Results
• Conclusions
DPLL Adaptive Algorithm

- Sense number of consecutive UP or DN pulses, $N_{err}$:
  - If so, then increase size of error by factor of 2, then gradually decrease
- Sense if have a number of consecutive overflows, $N_{cerr}$, or if can’t get back to error size of 1
  - If so, then increase value of $K_R$ to “ride out” the deterministic jitter

- Values of $N_{err}$ and $N_{cerr}$ are determined empirically and depend on how much noise is expected to be injected into DPLL (can set by software)
- Value of $K_i$ greatly affects loop bandwidth and is adjusted to reduce ICO jitter, which can be determined at time of design
Outline

• Introduction & Motivation
• Digital PLL Architectures
• Proposed DPLL Architecture
• Analysis of DPLL
• DPLL Adaptive Algorithm
• DPLL Circuit Implementation
• Measurement Results
• Conclusions
DPLL Circuit Implementation

- **10-bit Current DAC:**
  - MSB is single-ended switches $\rightarrow$ zero current when off
  - LSB is current steering switches $\rightarrow$ minimum glitch energy when switching
  - All Isrc are cascoded, thermally decoded and common centroid for best resolution at low currents

- **5 MSB, 5 LSB**
DPLL Circuit Implementation

• Corner cases:
  – If LSB word is 00000₂ or 11111₂ or near these 2 words, an MSB can toggle

• Case 1:
  – If LSB > 11000₂, subtract 1000₂ and turn on a special ½MSB PMOS current source

• Case 2:
  – If LSB < 01000₂, add 1000₂ and turn off another special ½MSB PMOS current source that is normally always turned on

• Both special current sources have current steering switches
DPLL Circuit Implementation

- **ICO:**
  - Three stage ring oscillator with Maneatis symmetric loads
- **Current Source:**
  - Constant-gm current source matched to the ICO delay cell
- **Digital Filter:**
  - CLA adders implemented for minimum phase margin degradation due to loop filter delay
  - $K_I$ and $K_R$ restricted to powers of 2
  - Fast logic elsewhere
DPLL Circuit Implementation

- Effect of Loop Filter delay on jitter:
  - Increased jitter due to:
    - Reduced phase margin
    - Missing of edges at PFD due to delay of “completion” signal
Outline

• Introduction & Motivation
• Digital PLL Architectures
• Proposed DPLL Architecture
• Analysis of DPLL
• DPLL Adaptive Algorithm
• DPLL Circuit Implementation
• Measurement Results
• Conclusions
Experimental Results

- Phase Noise:
  - -87dBc/Hz
  - Closed loop BW ~ 2MHz
Experimental Results

- Jitter Histogram Plot:
  - Peak-to-peak jitter is 270ps at 144MHz
  - Distribution is fairly Gaussian indicating low quantization phase errors
  - Very stable performance over temperature
Experimental Results

- Summary of Performance:

<table>
<thead>
<tr>
<th>Technology</th>
<th>0.25um CMOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power Supply</td>
<td>2.6V</td>
</tr>
<tr>
<td>Power Consumption</td>
<td>3.12mW @ 144MHz</td>
</tr>
<tr>
<td>VCO Range</td>
<td>40MHz – 160MHz</td>
</tr>
<tr>
<td>Peak-to-Peak Jitter</td>
<td>270ps @ 144MHz</td>
</tr>
<tr>
<td>Rms Jitter</td>
<td>60ps @ 144MHz</td>
</tr>
<tr>
<td>Cycle Jitter (T_{avg}-T_{min})</td>
<td>130ps @ 144MHz</td>
</tr>
</tbody>
</table>
Comparison with State-of-the-Art

- Figure-of-Merit: \[ \text{FOM} = \left[ \frac{\text{area}(\text{mm}^2)}{\left(\frac{\text{tech}}{0.25}\right)^2} \right] \left[ \frac{\text{mW}}{\text{MHz}} \right]^{1.5} \left[ \text{jitter(ps)} \cdot \sqrt{\text{mW}} \right]^2 \]

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Area</td>
<td>1</td>
<td>2.79</td>
<td>1.113</td>
<td>3.43</td>
<td>0.64</td>
</tr>
<tr>
<td>Pwr</td>
<td>1</td>
<td>9.23</td>
<td>12.11</td>
<td>2.30</td>
<td>5.77</td>
</tr>
<tr>
<td>T_L</td>
<td>1</td>
<td>0.61</td>
<td>0.085</td>
<td>3.33</td>
<td>2.40</td>
</tr>
<tr>
<td>Jitter</td>
<td>1</td>
<td>1.52</td>
<td>3.324</td>
<td>1.22</td>
<td>1.03</td>
</tr>
<tr>
<td>FOM</td>
<td>1</td>
<td>182.3</td>
<td>518.8</td>
<td>17.74</td>
<td>9.46</td>
</tr>
</tbody>
</table>
Conclusions

- A fast-lock, low-jitter Digital PLL has been proposed
- Jitter is reduced by using an adaptive bandwidth algorithm, high-resolution iDAC
- Low-power achieved by elimination of extra overhead circuitry such as voltage-to-current (V2I) converter and charge pump and well partitioned iDAC
- Area has been reduced by digitizing the loop filter
- Proposed Digital PLL has very stable performance over corners due to elimination of much of analog circuitry