Performance Bounds of Asynchronous Circuits with Mode-Based Conditional Behavior

Mehrdad Najibi       Peter A. Beerel

18th IEEE International Symposium on Asynchronous Circuits and Systems
Talk Outline

• Context and Motivation
  • Slack Matching and Conditional Circuits

• Previous Work
  • Performance analysis and Slack Matching

• Mode-Based Problem Statement
  • Intuitive introduction and Petri net formalism of modes

• Proof Technique and The Bound
  • Super-segments and their application to conditional slack matching

• Summary and Future Work
Motivation - Async Pipelines and Slack Matching

The Slack Matching Problem - Add minimum number of pipeline buffers to the circuit to meet a target cycle time $\tau$.

- This problem is unique to asynchronous design
- Unfortunately, often adds up to 30% area and power

Peter A. Beerel; Andrew M. Lines; et. al., “Slack matching asynchronous designs,” ASYNC’06
Motivation – Conditional Communication

Conditional communication reduces token flow, saving power

- Traditionally - manually introduced via user-created decomposition
- Recent research - automatically introduced via Operand Isolation

Arash Saifhashemi, Peter A. Beerel, “Automatic Operand Isolation in High-Throughput Asynchronous Pipelines,” to be submitted, PATMOS’12
Previous Works
Performance Bounds

Unconditional Circuits

- Throughput bounds – importance of bubbles [Greenstreet‘90]
- Analysis of Meshes [Pang’97]
- Canopy Graphs [Williams’91, Lines’98]
- Bottleneck Analysis [Taubin’09]
- Time Separation of Events [Hulgaard’93, Chakraborty’01]
- Variable delays [Yahya’07]

Conditional Circuits

- Xie and Beerel – Markovian (1997) and Monte-Carlo (1998) Analysis
- Canopy Graph Based Estimation [Gill’08]

None yield closed-form performance bound for conditional circuits
Previous Work
Slack-Matching

Unconditional Circuits
• MILP/LP formulation [Beerel’06, Prakash’06]

Conditional Circuits
• Bottleneck Removal Approaches [Gill’09]
  • Unfortunately, cannot give guaranteed performance
• Heuristic Iterative Algorithms [Venkataramani’06]
  • Simulation-based performance guarantees
• Industry approach [Beerel’11]
  • Treat conditional circuit as unconditional – ignore conditionality
  • We believe that this is conservative – but no proof given (till now)!
Find an upper bound on the average cycle time of the circuit given:

- Frequency of each mode
- Cycle time of each mode
- Unknown mode order
The Core Idea

Impact of mode change spans multiple \((k)\) segments, i.e., cycles – this paper bounds \(k\)
Performance Model

- **Petri-Nets:** \( \langle P, T, F, M_0 \rangle \)  
  \[ F \subset P \times T \cup T \times P \]  
  \[ M_0 : P \rightarrow \mathbb{N} \]
  - Places are annotated with delay values
  - Choices model conditionality
Example: Modeling Async Circuits using Petri-Nets

Full Buffer Channel Net (FBCN)
Elevation - Proof Technique
Super-Segments

\[ \text{Delay(cycle)} \leq 5 \tau_{\text{Elevated}} \]
Theorem: The average cycle time of the conditional Petri-net is bounded by the cycle time of the maximum super-segment.
Definitions

• Time Separation of Events

\[ \gamma_U^{(k)}(s, t, \epsilon) = \tau(t^{(k+\epsilon)}) - \tau(s^{(k)}) \]

• Average Cycle Time

\[ \tilde{\gamma}_U(t) = \lim_{n \to \infty} \frac{\gamma^{(0)}(t, t, n)}{n} \]
Assumptions to Derive the Bound

- Frequency of modes is known
  - The exact sequence of modes is not known
- Petri-Net of the circuit has the following properties
  - Safe & Live
  - Reversible
  - Unique–Choice
  - A reachable marking exists which marks all the simple cycles of the Petri-Net.
- Super-segment cycle times are known
Bound Formulation

\[
\bar{\gamma}_U(t) \leq \sum_{j=1}^{m} \hat{f}_j \cdot \tau(s_j^*)
\]

\[
\begin{align*}
\hat{f}_m &= \text{Min}(f_m \cdot \kappa, 1) \\
\hat{f}_j &= \text{Min}(f_j \cdot \kappa, 1 - \sum_{k=j+1}^{m} \hat{f}_k), \quad 1 \leq j < m
\end{align*}
\]

- \( f_j \): original frequency of the \( j^{th} \) mode
- \( \tau(s_j^*) \): cycle time of the \( j^{th} \) super-segment
- \( \hat{f}_j \): frequency of the \( j^{th} \) super-segment, post elevation
- \( \kappa \): maximum number of tokens in a place-simple cycle
Proof: Step 1

Known mode sequence: Cycle extraction

Modes: \( m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, m_9, m_{10} \)

Segments: \( s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10} \)

Cycle Times: \( \tau_1 \geq \tau_2 \geq \tau_3 \geq \tau_4 \geq \tau_5 \geq \tau_6 \geq \tau_7 \geq \tau_8 \geq \tau_9 \geq \tau_{10} \)

Super-segments: \( s^*_1, s^*_2, s^*_3, s^*_4, s^*_5, s^*_6, s^*_7, s^*_8, s^*_9, s^*_{10} \)

Elevated CT: \( \tau^*_1 \geq \tau^*_2 \geq \tau^*_3 \geq \tau^*_4 \geq \tau^*_5 \geq \tau^*_6 \geq \tau^*_7 \geq \tau^*_8 \geq \tau^*_9 \geq \tau^*_ {10} \)

\( \kappa = 3 \)
Proof Step 2: Unknown mode sequence

- Worst Case Mode Sequence
  - Results in longest critical cycle
  - Cycle extraction on worst case mode sequence results in the proposed bound

Segments:

Elevated CT:

\[ \tau_1^* \geq \tau_2^* \geq \tau_3^* \geq \tau_4^* \geq \tau_5^* \geq \tau_6^* \geq \tau_7^* \geq \tau_8^* \geq \tau_9^* \geq \tau_{10}^* \]

Distributing slowest modes once per \( \kappa \) segments yields worst case
Slack-matching Using The Bound
- A Simple Example

Suppose there are two modes of operation

- “Slow” Mode $s_1$ – Slack matched to 36 transitions per cycle
  - Mode 1 is rare – 1% activity
- “Fast” Mode $s_2$ – Slack matched to 18 transitions per cycle
  - Max tokens in place: simple cycle $\kappa$ of super-segment $s^*_{1}$ is 10
  - The resulting bound is $18 \times 0.9 + 36 \times 0.1 = 19.8$

If performance bound not good enough

- Slack match slow mode $s_1$ to 22.5
- The resulting bound is 18.4

Yields lower area/power than slack matching as if unconditional
Summary and Conclusions

This paper presents several firsts

• First closed-form formula that bounds performance of conditional asynchronous circuits

• First proof that slack-matching conditional circuits unconditionally is conservative

• First performance-driven conditional slack-matching algorithm that saves area and power over unconditional slack matching

This paper provides useful intuition

• We can characterize the performance of a conditional circuit using marked graphs that describe their modes of operation

• Each mode change impacts a bounded number of segments
  • But, if not otherwise constrained, the bound is relatively large