

## 動的タイムボローイングを可能にするクロッキング方式

吉田 宗史† 有馬 慧‡ 岡田 崇志‡ 塩谷 亮太‡ 五島 正裕‡ 坂井 修一‡

† 東京大学工学部 電子情報工学科

‡ 東京大学大学院 情報理工学系研究科

### 1 背景

近年の半導体プロセスの微細化に伴い、トランジスタや配線のサイズが原子のサイズに近づき、素子遅延のチップ内のランダムなばらつきが大きな問題となっており、ばらつきが増大していくと、従来のワースト値に基づいた設計手法は悲観的になりすぎる(図1)。そのため、ワースト・ケースより実際に近い遅延(ティピカル値)に基づいた動作を実現する手法が提案されている。

#### 動的タイミング・フォールト検出・回復

タイミング・フォールトは、遅延の動的な変化によって設計者の意図とは異なる動作が引き起こされる過渡故障である。以下、タイミング・フォールトはTFと略す。動作時にTFを検出し、回復する動的な手法も、ばらつき対策として有効であり、Razor[1]は、特殊なFFを用いてTFを動的に検出し、回復)する。このような回路にDVFS(Dynamic Voltage and Frequency Scaling)を組み合わせると、見積もりではない、その個体のその時の動作環境における実際の遅延に応じた電圧と周波数で動作を実現することができる。

#### 入力ばらつき

それと同時にこの手法では、ロジックの遅延の入力ばらつきを部分的に利用することができる。

ロジックの遅延の入力ばらつきとは、ロジックの実効的な遅延が入力に依存することである。例えば多ビットの加算器では、桁上がりの伝搬によって最上位ビットが反転するような入力が与えられた時に、出力が確定するまでの時間が最大となる。この時、ロジックの

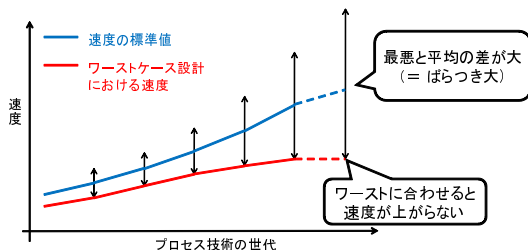


図1: ばらつきによるティピカル値とワースト値の乖離

†Shuji YOSHIDA ‡Satoshi ARIMA ‡Takashi OKADA ‡Ryota SHIOYA ‡Masahiro GOSHIMA ‡Syuichi SAKAI  
†Dept. of EEIC Eng, the Univ. of Tokyo  
‡Dept. of Information and Communication Eng, the Univ. of Tokyo

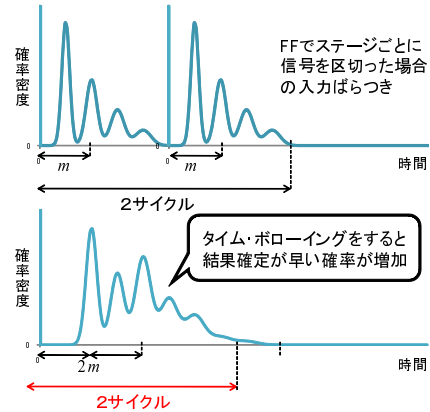


図2: タイムボローイングをした場合の入力ばらつき

クリティカル・パスが活性化されたと言い、ロジックのワースト遅延とは通常クリティカル・パスのワースト遅延のことを言う。動的TF検出を行うと、クリティカル・パスが活性化された時にそれを確実にTFとして検出できるという条件を満たせば、クリティカル・パスの遅延を下回る動作周期でも動作させることができる。

### 2 提案手法

#### 本稿の提案：動的タイムボローイング

大数の法則が示すように、ゲート段数を増加していくと、パスの遅延は構成する個々のゲートのティピカル遅延の総和に近づく。すなわち、遅延のばらつきを統計的に扱う(図2)ことで、パスが十分に多段であれば、ばらつきの影響は無視できるようになるのである。

そのためには、前後のステージ間で時間を融通(タイムボローイング)し、各ステージごとに課されているタイミング制約を緩める必要がある。

本稿で提案するのは、端的に言えば、TF検出と二相ラッチを組み合わせたクロッキングの方式である。本手法により、動的タイムボローイングが可能になる。従来からある二相ラッチ方式で可能になるタイムボローイングは、設計時におけるもので、言わば静的タイムボローイングと呼ぶことができる。それに対して、本手法で可能になる動的タイムボローイングとは、以下のようなものである(図3)。あるステージの遅延がクロック周期を超えることを遅延の「借金」と考えると、

これらの方式は、以下のように説明することができる：

**従来の方式** 従来のワースト・ケースに基づくクロッキング方式は、言わば、借金の概念がない方式である。借金が生じたことを検出することができないため、万一生じれば、いわゆる silent error となって極めて問題である。そのため、借金が生じる確率が限りなく 0 に近くなるようにワースト・ケース設計を行う必要がある。

**TF 検出** TF 検出は、借金は許さないが検出はできる方式と言える。借金が生じれば、検出される。その後「破綻処理」を行い、動作を再開することになる。

**提案方式** 提案する手法は、ある程度の借金を許す方式であると言える。提案方式では、あるステージで生じた借金を次のステージで「返済」することができる。1 回の借金ですぐさま破綻することはない。連続して赤字が続き、借金がかさんだ時にはじめて破綻処理を行い、動作を再開する。

**回路構成と動作**

二相ラッチでは半 FF ロジック区間あたりの遅延の上限は 1 サイクルであったが、提案手法 (図 4) ではショート・パスに遅延素子を挿入し、TF 検出で正しい値をサンプリングできるようにすることで、最大遅延制約を 1.5 サイクルまで拡大している。

遅延を挿入したことによる遅延大への偏りを補うために、ロジックの出力を予測する回路 (Pred) が付加されており、これをセレクタで切り替えながら次段に流す。予測のヒット率が高ければ、入力ばらつきはさらに遅延小に偏る。

これによって遅延 (借金) が累積しても TF が発生しにくくなると同時に、予測回路による借金返済のチャンスも増えている。現在、64bit のリプルキャリーアダーによるカウンタを基に、提案手法の回路モデルの実装・評価を行っている。(図 5)

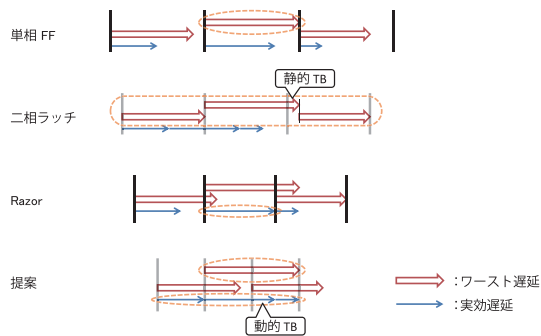


図 3: 各種クロッキング方式との比較

2相ラッチ ※ Short < 0.5 < Critical

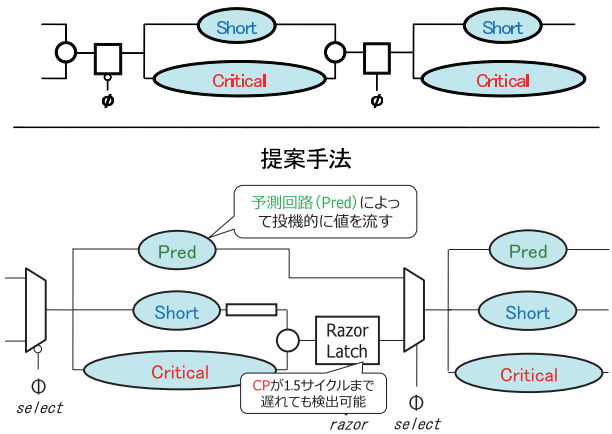


図 4: 二相ラッチ (上) と提案手法の回路 (下)

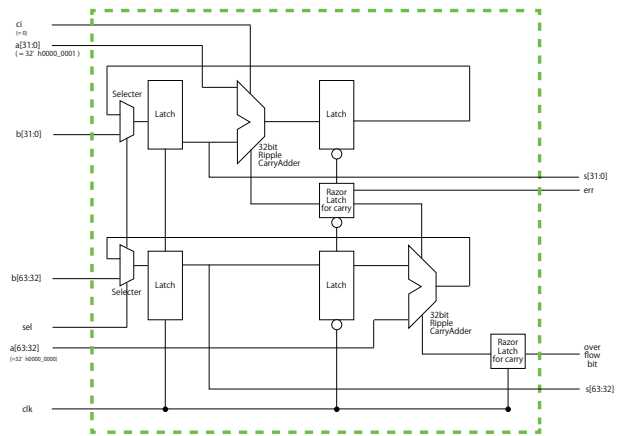


図 5: 64bit カウンタを用いた提案回路モデルの実装

**まとめ**

製造ばらつき対策手法は数多く提案されているが、実は製造ばらつきによる遅延の変動は、入力ばらつきのそれに比べればとても小さい。提案手法は遅延が大きくばらつくこと前提として、逆に変動を利用する手法となっているため、ばらつきを吸収して高クロック化や低電圧化を達成できる。

**参考文献**

[1] D. Blaauw, S. Kalaiselvan, K. Lai, Wei-Hsiang Ma, S. Pant, C. Tokunaga, S. Das, and D Bull. Razor II: In Situ Error Detection and Correction for PVT and SER Tolerance. In *Int'l Symp. on Solid-State Circuits Conference (ISSCC)*, 2008.

[2] 喜多 貴信. タイミング制約を緩和するクロッキング方式. 東京大学修士論文, 2010.