

タイミングエラー見逃しを緩和するカナリアフリップフロップの提案と評価

澤田 颯斗^{1,a)} 小松 聡^{1,b)}

概要: 本研究ではカナリアフリップフロップ (以下カナリア FF) の欠点を補う改良型カナリア FF を提案し、回路の信頼性を高めることを目的としている。近年、高速動作と低消費電力を同時に満たす回路の需要が高まっている。しかし所望の回路動作を保証するためには、電源電圧や周波数に大きなマージンが必要となる。これらのマージンを削減するために、タイミングエラーを予告するカナリア FF と、電圧や周波数を動的に制御する DVFS を組み合わせた手法が提案されている。本研究ではカナリア FF の機能を延長し、メイン FF のタイミングエラーを緩和する新たなカナリア FF を提案し、回路に実装した時の信頼性向上の検証を行う。

Propose and Evaluation of New Canary Flip Flop to Mitigate the Missing Timing Error Prediction

SAWADA HAYATO^{1,a)} KOMATSU SATOSHI^{1,b)}

Abstract: This work, we propose the new Canary Flip-Flop (hereinafter, referred to as "Canary FF") which is compensated for the weakness of the Canary FF and we aim at improvement of reliability of circuits. Recent years, low power consumption and fast operation circuits are needed, but to guarantee correct operation, circuit designers must prepare the voltage and frequency margin. For decreasing these margin, the method which combines Canary FF with DVFS has been proposed. We propose the new Canary FF which are combined conventional Canary FF and Soft-Edge FF, and we are evaluating improvement of reliability.

Keywords: Canary FF, Standard Cell, Synthesis, Reliability

1. はじめに

近年の半導体製造プロセスの進歩により、トランジスタの微細化が急速に進み、高速かつ低消費電力な集積回路が実現可能となってきている。また、近年注目が集まっている IoT によって、今後様々な機能を実現するために集積回路の需要は増加すると予想される。そのため、製造時の素子特性ばらつきやノイズ、温度変化などを考慮して電力・性能マージンを確保する必要があるが、これらは同時に回

路の機能や性能を著しく制限してしまうことがある。このような設計マージンを削減するためにいくつかの手法が提案されており、本研究ではその中でもカナリアフリップフロップ (FF) [1], [2], [3], [4] に着目した。カナリア FF を、電源電圧や周波数を適応的に変化させる Dynamic Voltage and Frequency Scaling (DVFS) [5] と合わせて使用することで、製造ばらつきや外的要因による遅延変動に対応し、効率よく電力・性能マージンを削減することができる。

本研究では、過去に提案されたカナリア FF の機能を延長し、タイミングエラー発生確率を緩和することで、回路の信頼性をより向上させることを目的とし、面積、電力オーバーヘッドやエラー率などの評価を行う。また本稿では、カナリア FF を導入する際の、設計フローへの導入容易性についても述べる。

¹ 東京電機大学 工学研究科 電気電子工学専攻
Electrical and Electronic Engineering
Graduate School of Engineering
Tokyo Denki University

a) 17kmh10@ms.dendai.ac.jp

b) komatsu@mail.dendai.ac.jp

本稿の構成を以下に示す。2章ではカナリア FF のや Razor FF など関連する研究を紹介する。3章ではカナリア FF の原理や長所・短所について述べる。4章では3章で紹介するカナリア FF の特徴をもとに本研究で提案する新しいカナリア FF について述べる。5章ではカナリア FF をどのように設計フローに組み込むかを示す。6章では従来のカナリア FF との比較方法を述べ、7章では現在までの結果を示す。最後に8章でまとめとする。

2. 関連研究

カナリア FF をはじめとする様々な FF・ラッチベースの手法は、それ以前に提案された Razor FF[6] を元としている。Razor FF の回路図を図1に示す。この手法は回路動作に使用する通常の FF(メイン FF) と並列に通常の FF やラッチ(シャドウ FF・ラッチ)を接続し、シャドウ側にはわずかに位相が遅れたクロック(図1中の Clk_del)を入力させる。これにより、通常のクロックタイミングにメイン FF の D 入力に信号が間に合わなかった場合、遅れてサンプリングしたシャドウ側の FF が本来の値を保存する。この2つの FF が保持した値を XOR ゲートで比較し、値が異なればタイミングエラーを起こしている、として図中の ERROR からエラー検出信号が出力される。タイミングエラーが検出された場合、回路動作を停止させ、エラー訂正回路などにより、正しい信号を復元することでエラーから回復する。通常、回路を設計する時点で、タイミングエラーを引き起こさないように電源電圧や周波数が決定されるが、このエラー検出手法と DVFS(DVS) を組み合わせることで必要以上に確保されたマージンを減らすことができ、電力や性能の効率を上げることができる。Razor FF を用いたマージン削減手法では、エラー回数がある決められた閾値以下であれば電源電圧を降圧、または周波数を上げ、閾値以上であれば電源電圧を昇圧、または周波数を下げる。シャドウ側に追加されたラッチや、値を比較する XOR ゲートによるオーバーヘッドはあるものの、回路を比較的簡単に構成でき通常動作時の電力オーバーヘッドは約3%であると報告されている [6]。

しかしこの Razor FF にはいくつかの欠点もある。まず、エラーを訂正する機構を備えるプロセッサのような回路を想定しているため、それ以外のクロック同期回路には導入しにくく、訂正回路によって面積・電力オーバーヘッドが増加する。またシャドウ側の最小パス遅延違反によるエラー検知(ショートパス問題)を起こす可能性があり、ショートパスに遅延バッファを挿入するなど回路自体の設計を、気をつけて行わなければならない。そして遅延クロックをグローバルで生成するか、ローカルで生成するかも議論が必要であり、グローバルで生成する場合はメイン側のクロックとの間のスキューに気をつけなければならない。ローカルで生成する場合は遅延バッファによる面積・電力の増

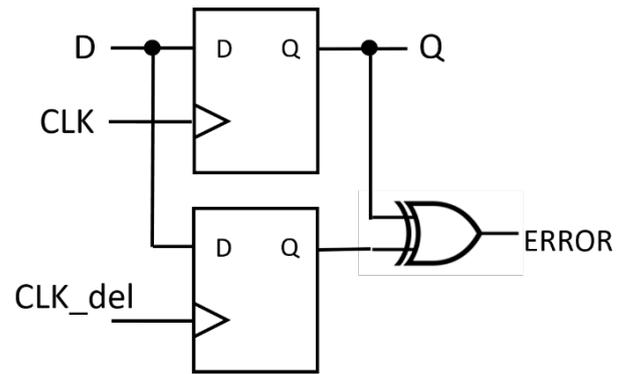


図1 Razor FF

Fig. 1 Razor FF

加が顕著になる可能性がある。

本研究の要となるカナリア FF[1], [2], [3], [4] は、Razor FF の改良版として提案され、主にタイミングエラーを予告する目的で使用される。カナリア FF の回路図を図2に示す。動作原理は Razor FF と同様にダブルサンプリング方式を採用している。ここで注目すべきは、シャドウ側の信号に対して遅延させているのは、クロック信号ではなく D 入力としている点である。これにより遅延クロックを生成する必要がないので通常のクロックドメインと遅延クロックドメイン間のスキューが発生しないので、Razor FF で問題となっていたシャドウ側のショートパスによるエラー誤検知を防ぐことができる。また、シャドウ側の D 入力遅延は、製造時にどのようにばらついても常にシャドウ側がメイン側より先にタイミングエラーを起こす。したがってタイミングエラーを予告する機能は保証されている。そして Razor FF と同様に回路構成が比較的単純であり、またエラー訂正回路の導入も必要ないため、あらゆる回路への導入に対する敷居が低いのも特徴として挙げられる。

カナリア FF の注目すべき欠点は、メイン FF のタイミングエラーを 100%避けることができない点である。そのためシャドウラッチの D 入力は遅延を大きくしなければならず、エラーが予告されるたびに電圧を昇圧、あるいは周波数を下げる必要があり、マージン削減が保守的になる。

ソフトエッジ FF[7] は、マスタースレーブ (MS) 型 FF の MS 間クロックスキューを逆手に取り、両ラッチが同時に開口している時間を作ることでわずかなタイムボローイングを行い、製造ばらつきやノイズなどの遅延変動によるタイミングエラーを緩和する。ソフトエッジ FF の回路図を図3(a)に、クロック回路を図3(b)にそれぞれ示す。通常の回路の最小パス違反を起こさない程度にマスタのクロック入力を遅らせることで微小遅延変動に対応でき、通常的设计フローをあまり変更することなく導入できる。ホールド時間の制約を考えなければならないが Razor FF ほど大きく遅延を起こすことがないので、導入の敷居は低いと考えられる。消費電力のオーバーヘッドはわずか 0.3~2.8%と

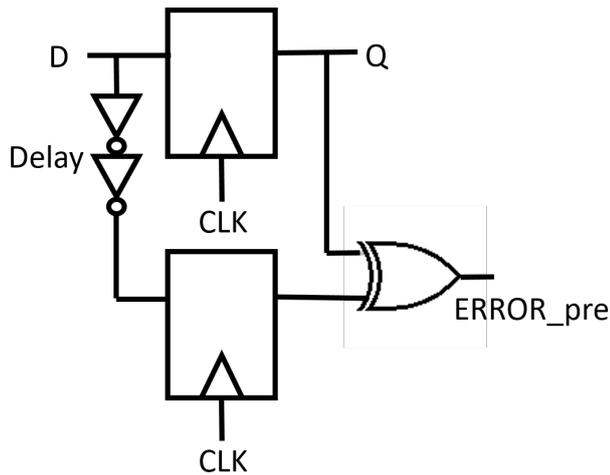


図 2 カナリア FF
Fig. 2 Canary FF

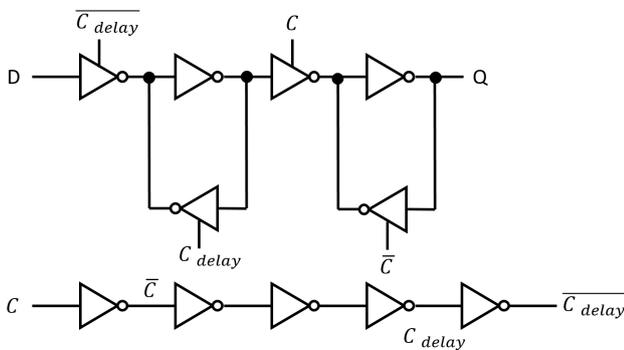


図 3 ソフトエッジ FF(a)FF 単体 (b)MS クロック回路
Fig. 3 Soft-edge FF(a)FF only (b)MS Clock Circuit

なっている。欠点は、MS 両方が開口している時間がホールド時間増大の原因となってしまうことである。FF においてホールド遅延が増加すると、クロックスキュー耐性が下がるので、やはり設計時に気を付ける必要がある。

このほか、Razor FF のようにタイミングエラーの検出・訂正を行う Razor-Lite[8] や、タイムBORROWINGを利用してタイミングエラーを起こすことなく回路を動かすことができる TIMBER FF[9] や Transition Detector with Time Borrowing(TDTB)・Double-Sampling with Time Borrowing(DSTB)[10] も提案されている。

3. 提案カナリア FF

[2] によると、急な遅延変動はないとすれば、メイン FF のタイミングエラーは起きないと述べられている。しかしノイズや温度変化は容易に発生し得るので、より信頼性を高めるには外的要因による遅延増加にも対処する必要がある。本研究ではカナリア FF の「メイン FF のタイミングエラーの問題」と、それに起因する「保守的なマージン削減」についての改善を目指す。上記 2 点の問題の原因は共通して「メイン FF のタイミングエラー」である。そこで

メイン FF のタイミングエラーを防ぐために、メイン FF をソフトエッジ FF にすることで、シャドウのタイミング制約を厳しいままメイン FF のパスに余裕を持たせる、改良型カナリア FF(以下提案カナリア FF)を提案する。回路図を図 4 に示す。メイン側にはソフトエッジ FF を採用し、シャドウ側にはラッチを採用している。

タイミングエラーと予報が起きていない通常動作時のタイミングチャートを図 5 に示す。面積削減のため、提案カナリア FF ではシャドウ側にはラッチを使用しているが、動作原理をわかりやすくするため図 5 においてはシャドウ側に FF を使用したと仮定して描画した。実際の回路において、ラッチではエラー予報信号は半クロックしか保持できないので、OR ツリーでエラー予報信号を収集したのち、ラッチで残り半分のクロックサイクル中値を保持するようにしている。それぞれの信号名は図 4 と対応している。 C は元になるクロック、 $\overline{C_{delay}}$ は C からわずかに遅れたメイン FF のマスタ (M) のラッチを制御するクロック、 D は FF のデータ入力、 D_{shadow} はシャドウラッチに入力される Delay 分だけ遅延させたデータ入力、 Q はメイン FF の出力、 Q_{shadow} はシャドウ側の出力、ERROR_pre はエラー予報信号を示す。パス遅延がクロックサイクル中に十分余裕があればメイン FF、シャドウ FF(ラッチ) はともに所望の信号を保持する。

エラー予報時のタイミングチャートを図 6 に示す。シャドウ側の入力は Delay 分遅れるので、パス遅延に余裕がなくなるとシャドウ FF(ラッチ) がタイミングエラーを起こす。この図の場合はメイン FF はタイミングエラーを起こしていないので、 Q と Q_{shadow} を比較して ERROR_pre からタイミングエラー予報が出力される。[1], [2], [3], [4] の DVS 方式によれば、この時点で電圧や周波数の変更が必要となる。

エラーマスク時のタイミングチャートを図 7 に示す。 C の立上りエッジに D 入力が間に合っておらず、本来であれば Q と Q_{shadow} が L を出力する。しかし提案カナリア FF ではメイン FF の M ラッチに入力されるクロックがスレーブ (S) ラッチよりも遅くなっているため、M と S のラッチがともに開口する時間ができる。これによりタイムBORROWINGが発生し、M ラッチが閉口するまでの間、入力の変化を許容することでタイミングエラーをマスクしている。ここで注意しなければならないのは、タイムBORROWINGした分だけ次ステージのタイミング制約が厳しくなってしまうことである。そのため長いパスが続く可能性はとても低いという前提 [9] での提案となる。

提案カナリア FF の欠点はソフトエッジ FF と同様にホールド時間が伸びることである。論理合成ツールはライブラリにあらかじめ登録された FF のホールド時間を元に、回路中に自動でバッファを挿入する。挿入されたバッファによる面積・電力の比較は今後いくつかの回路を対象として

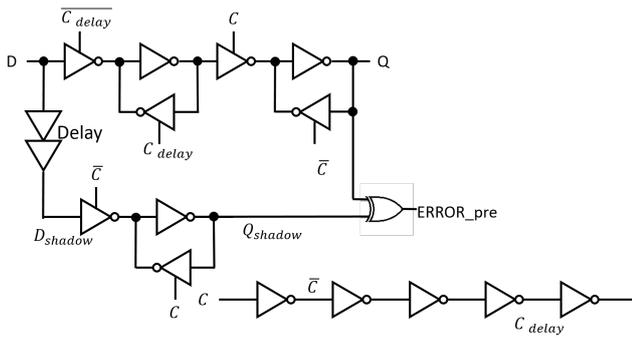


図 4 提案カナリア FF

Fig. 4 Propose Canary FF

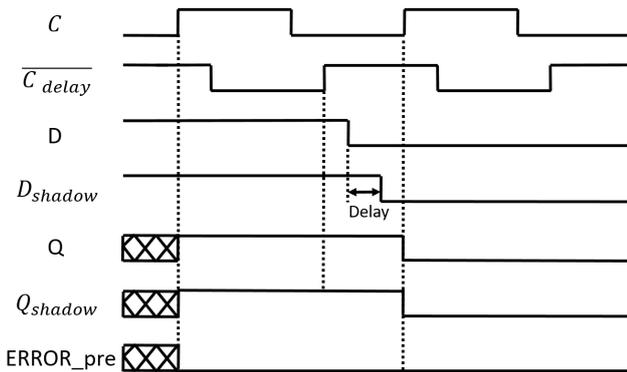


図 5 提案カナリア FF のタイミングチャート (正常時)

Fig. 5 Timing Chart of Propose Canary FF (at Normal)

検証する予定である。

電源電圧を変更するアルゴリズムとして [1], [2], [3], [4] では、エラーが予告された時点で昇圧する方法が提案されている。これはメイン FF のエラーを防ぐためである。しかしこの方法では保守的になり、積極的な電源電圧の降圧が課題であることが示唆されている [2]。本研究の提案カナリアはメイン FF のエラーを防ぐように働く。そこで本研究では提案カナリアを利用した積極的な降圧アルゴリズムとして、Razor FF と同様の予告回数と閾値を比較する方法を提案する。依然として Razor FF よりはエラーを拾う能力に劣るため閾値は低めに設定する必要があるが従来手法より積極的な電圧の降圧ができ、電力マージンの削減ができることが期待できる。

4. 提案カナリア FF の設計フローへの導入

本章ではカナリア FF 及び提案カナリア FF を導入した設計フローについて述べる。回路設計フローを図 8 に示す。破線で示した部分は通常的设计フローを、実線部分はカナリア FF を導入する際に追加となる手順を示す。スタンダードセルライブラリは一度登録しておけば次回以降も使用できるため、カナリア FF 自体の設計はセットアップ・ホールド時間や予告開始時間等の仕様を決めておけば一度でよい。なお本研究の環境は、論理合成用のライブ

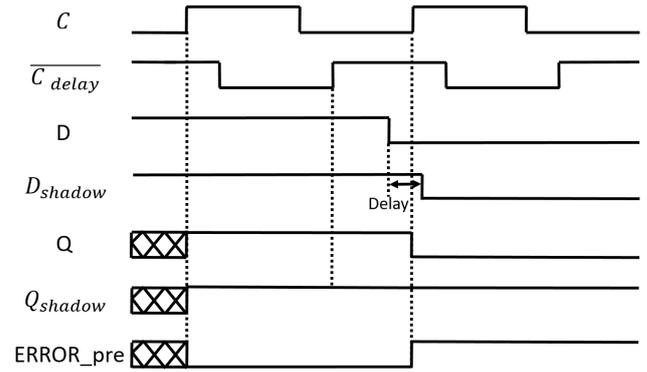


図 6 提案カナリア FF のタイミングチャート (予告時)

Fig. 6 Timing Chart of Propose Canary FF (at Predicting Timing Error)

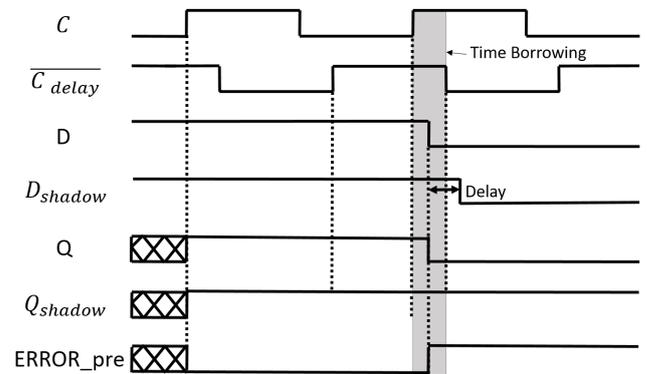


図 7 提案カナリア FF のタイミングチャート (エラーカバー時)

Fig. 7 Timing Chart of Propose Canary FF (at Covering Timing Error)

ラリ生成には Synopsys 社の SiliconSmart を、配置配線用のライブラリ生成には Synopsys 社の Milkyway を、論理シミュレーションには Cadence 社の Verilog-XL を、論理合成には Synopsys 社の Design Compiler を、配置配線には Synopsys 社の IC Compiler を、カナリア FF のレイアウト設計は Cadence 社の Virtuoso をそれぞれ使用した。SiliconSmart でライブラリを登録する際、対象セルの論理を記述しなければならないが、シャドウ側に遅延を使用しているカナリア FF はロジックのみで記述することができない。幸いカナリア FF の論理的な機能は通常の FF と比較するとエラー予告信号以外同じである。そのため出力端子としてエラー予告信号端子を残したまま通常の FF として論理を記述することでスタンダードセル化した。この手順で生成したカナリア FF のライブラリを使用して論理シミュレーションをする場合は、Verilog-HDL でシャドウ側の D 遅延量とシャドウラッチ、出力値比較のための XOR を、論理シミュレーション用に生成されたカナリア FF の情報ファイルに追加で記述することで可能である。配置配線用のライブラリ生成には Milkyway に SiliconSmart で生成したライブラリ情報 (.db ファイル) と Virtuoso で設計したカナリア FF の GDS ファイルを入力することで生成

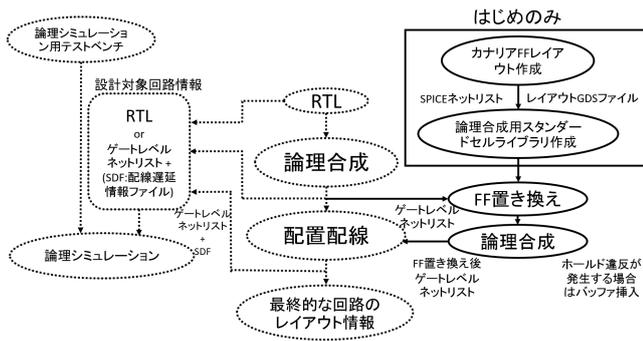


図 8 カナリア FF を導入したデザインフロー
Fig. 8 Circuit Design Flow using Canary FF

が可能である。カナリア FF への置き換えは、論理合成後のネットリストを書き換えるだけでよいので、セルのレイアウトができれば、論理合成ライブラリから置き換えまでのフローはスクリプトにより自動化が可能である。今後はスクリプトによって移植性向上を目指す予定である。

5. 実験結果

本研究の目的は、まずカナリア FF におけるメイン FF をタイミングエラーを起こしにくくし、かつ従来と同程度のエラー予告機能を維持することである。そこでスタンダードセルライブラリに含まれる FF、自作した従来のカナリア FF、提案カナリア FF の 3 種類についてセットアップ時間、ホールド時間、C-Q 遅延、エラー予告発生時間 (タイミングエラー予告がクロックエッジよりどれだけ前に開始されるか) を回路シミュレーションにより評価した。なお使用した標準スタンダードセルライブラリは ROHM 社 0.18 μm 京大版ライブラリ [11] であり、シミュレーションは Synopsys 社の HSPICE を用いた。FF の各特性を表 1 に示す。なお Setup はセットアップ時間、Hold はホールド時間、CQ delay は C-Q 遅延時間、PRE は予告開始時間 (クロックエッジよりどれだけ前に予告するか)、DFF は京大ライブラリの ROHM18DFP010、カナリアは従来のカナリア FF、提案は今回提案しているカナリア FF をそれぞれ示す。想定どおり提案カナリア FF の予告開始時間は従来のカナリア FF と同程度となり、ホールド時間が伸びていることが分かる。電源電圧を下げることでホールド時間も延長されるが、ホールド違反となる可能性のあるショートパスのゲート遅延も増加するので、定格電圧でショートパスによるホールド違反が起きなければ、電圧を降圧してもホールド違反は起きないと考えられる。

次にこれらを実際の回路に導入して、面積・電力オーバヘッドの評価を行った。今回導入の対象とした回路は 2 段パイプライン構成の 16 ビットリプルキャリア加算器とし、周波数 200 [MHz] の制約で論理合成・配置配線を行い、加算ロジック部分は同一のものとした。クリティカルパスでの検証は最悪ケースによる結果を評価するためであり、

表 1 FF 比較

Table 1 FF comparison

	Setup(rise) [ps]	Setup(fall) [ps]	Hold(rise) [ps]	Hold(fall) [ps]	CQ delay (rise) [ps]	CQ delay (fall) [ps]	PRE(rise) [ps]	PRE(fall) [ps]
DFF	45	105	-40	-60	277	237	-	-
カナリア	35	65	-20	-30	287	233	225	355
提案	-135	-5	140	70	351	272	225	355

ショートパスの評価は提案カナリア FF がホールド時間を伸ばしていることによる、意図しないホールド違反を起こしていないかを確認するためである。対象回路を図 9 に示す。入力は 16 ビットの A, B, C で、出力は 16 ビットの Y としており、エラーの予告はツリー状にした OR 回路から図の ERR_PRE として出力されるようになっている。この回路に対しクリティカルパスやショートパスを活性化するテストベクタを入力し、電源電圧を 1.8 [V] から 1.3 [V] まで 0.01 [V] 刻みで降圧した場合のエラー発生開始電圧と予告開始電圧をシミュレーションにより測定した。

ショートパス活性化の結果、1.3 [V] まで降圧してもホールド違反によるエラーは起きなかった。電源電圧と 1 段だけクリティカルパスが活性化される場合と、2 段連続でクリティカルパスが活性化される場合の消費電力を図 10 に、各 FF を導入した対象回路の面積比較を表 2 それぞれ示す。図 10 から提案カナリア FF を導入した加算器、従来のカナリア FF を導入した加算器、通常の FF を導入した加算器の順で電力が高いことが分かる。これはカナリア FF 及び提案カナリア FF におけるシャドウラッチや、エラー予告信号収集のための OR ツリー回路を追加したことによって、表 2 に示すように、トランジスタが増大したことが原因であると考えられる。また提案カナリア FF はクロック部にも遅延のためにインバータを追加しており、クロックによるスイッチングが頻繁に起こることも原因として挙げられる。カナリア FF と提案カナリア FF がエラーを予告し始める電圧とエラーを起こす電圧を表 3 に示す。エラー予告時間は同等であり、表 3 でエラーを起こすまでの電圧差が増加していることから、提案カナリア FF はホールド時間を延長することで本来エラーを起こす電圧でもエラーを防いでいることが分かる。ここで、ホールド時間をより長くするようにバッファを挿入すれば、より低い電圧で、パス遅延が増大したとしてもエラーを防ぐことができると考えられるが、先の結果から面積や電力増加の原因となり得るので、低消費電力かつ省面積で遅延を生み出す方法の検討が必要である。

6. まとめ

本研究では、メイン FF のタイミングエラーをより起こしにくくする改良型カナリア FF とその使用方法を提案し、容易な設計フローへの導入法も示した。結果から、本来タ

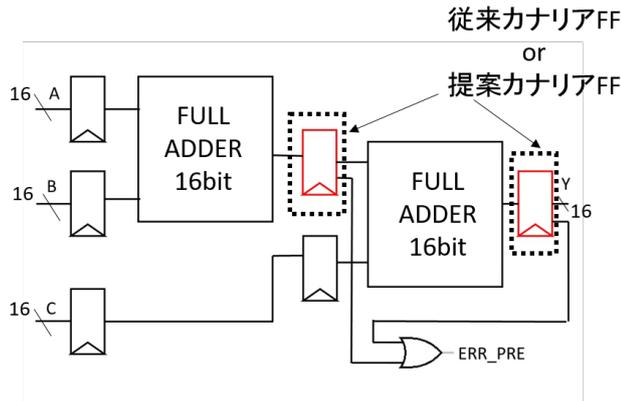


図 9 16 ビット加算器
Fig. 9 16bit Full Adder

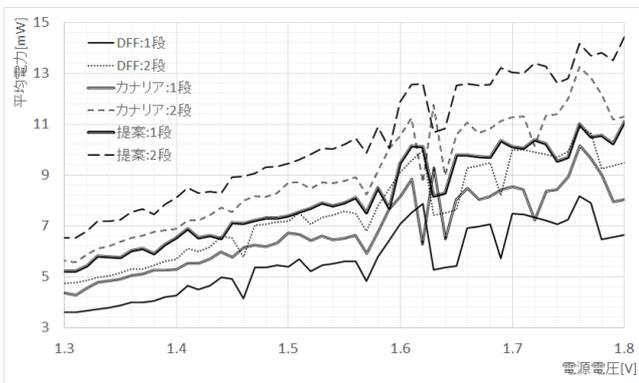


図 10 2 段パイプライン加算器の消費電力
Fig. 10 Energy Consumption of 2 Stage Adder

表 2 各 FF 導入時の面積比較

Table 2 Comparison of Adder Area with FFs

	総面積 [μm^2]	ORツリーの面積 [μm^2]	FFによる面積 差[μm^2]	通常の回路との 面積増加率[%]
DFF	30492	-	-	-
カナリア FF	35776	813	4438	17.3
提案カナ リアFF	38233	813	6916	25.4

表 3 エラー予告開始電圧とエラー開始電圧

Table 3 Highest Voltages of Error Occurance/Prediction

	1段のみクリティカルパス活性化			2段クリティカルパス活性化		
	エラー予告 開始電圧 [V]	エラー開始 電圧 [V]	予告からエラーを 起こすまでの 電圧差[V]	エラー予告 開始電圧[V]	エラー開始 電圧 [V]	予告からエラー を起こすまでの 電圧差[V]
カナリア FF	1.76	1.70	0.06	1.76	1.70	0.60
提案カ ナリアFF	1.75	1.65	1.00	1.80	1.70	1.00

イミューンエラーを起こすはずの電圧でもエラーを起こさないように働いていることが分かったが、提案カナリア FF を含んだ加算回路全体の電力は、通常の FF のものに比べて大きく増加してしまっただけで、提案したカナリア FF 自体の電力を削減する方法を検討する必要がある。また、FF や周辺回路を工夫することで信頼性の向上が見込める技術

は多数提案されているが、カナリア FF のように手軽に導入し信頼性を向上できるのは実用性で大きなアドバンテージであると言える。そのため、今後もカナリア FF を使用して電圧変更にかかる時間的・電力的オーバーヘッドを考慮した評価や、電圧変更アルゴリズムの比較を定量的に行っていきたいと考えている。

7. 謝辞

本研究は東京大学大規模集積システム設計教育センターを通じ、シノプシス株式会社、日本ケイデンス株式会社、ローム株式会社の協力で行われたものである。また本研究で使用したライブラリは京都大学情報学研究科田丸/小野寺研究室の成果によるものであり、京都工芸繊維大学小林和淑教授によりリリースされたものである。

参考文献

- [1] T.Sato and Y.Kunitake, "it A Simple Flip-Flop Circuit for Typical-Case Designs for DFM", 8th International Symposium on Quality Electronic Design,2007.
- [2] 佐藤寿倫, 国武勇次, "カナリア・フリップフロップを利用する DVS 方式の改良", 情報処理学会研究報告計算機アーキテクチャ,2007.
- [3] 佐藤寿倫, "カナリア・フリップフロップを利用する省電力マイクロプロセッサの評価", 先進的計算基盤シンポジウム (SACSIS),2007.
- [4] 佐藤寿倫, 国武勇次, "ばらつき耐性を持つカナリア FF を利用したデザインマージン削減による省電力化", 情報処理学会論文誌, Vol.49, No.6, 2029-2042, 2008.
- [5] T.Pering, T.Burd, and R.Brodersen, "The Simulataion and Evaluation of Dynamic Voltage Scaling Algorithms", International Symposium on Low Power Electronics and Design,1998.
- [6] D.Ernst,N.S.Kim,S.Das,S.Pant,R.Rao,T.Pharm,C.Ziesler, D.Blaauw,T.Austin,K.Flautner,and T.Mudge, "Razor: A Low-Power Pipeline Based on Circuit-Level Timing Speculation", 36th International Symposium on Microarchitecture,2003.
- [7] V.Joshi, D.Blaauw, and D.Sylvester, "Soft-edge flip-flops for improved timing yield: design and optimization", IEEE ACM International Conference, 2007.
- [8] I.Kwon, S.Kim, D.Fick, M.Kim, Yen-Po.Chen, and D.Sylvester, "Razor-Lite: A Light-Weight Register for Error Detection by Observing Virtual Supply Rails", IEEE Journal of Solid-State Circuits, Vol.49, No.9, 2014.
- [9] M.R. Choudhury, V.Chandra, R.C.Aitken, and K.Mohanram, "Time-Borrowing Circuit Designs and Hardware Prototyping for Timing Error Resilience", IEEE Transactions on Computers, Vol.63, No.2, 2014.
- [10] K.A.Bowman, J.W.Tschanz, N.S.Kim, J.C.Lee, C.B.Wilkerson, S-Lien L.Lu, T.Karnik, and V.K.De, "Energy-Efficient and Metastability-Immune Resilient Circuits for Dynamic Variation Tolerance", IEEE Journal of Solid-State Circuits, Vol.44, No.1, 2009.
- [11] 小野寺秀俊, 平田昭夫, 北村晃男, 小林和淑, 田丸啓吉, "P2Lib:スタンダードセルライブラリ自動生成システム", 情報処理学会論文誌, Vol.40, No.4, pp.1660-1669, 1999/04.