

# 90nm プロセス商用 FPGA にマッピングした リングオシレータの発振周波数の劣化評価

石井 翔平<sup>†</sup> 小林 和淑<sup>†,††</sup>

<sup>†</sup> 京都工芸繊維大学大学院工芸科学研究科 〒606-8585 京都市左京区松ヶ崎御所海道町  
<sup>††</sup> JST, CREST

あらまし 微細化に伴い顕著になってきた FPGA の経年劣化に関する問題に着目し、NBTI による FPGA の劣化を定量的に評価した。FPGA の経年劣化を測定する前に Cyclone II FPGA 上にリングオシレータをマッピングして、その発振周波数の分散を実測で測定した。測定の結果、リングオシレータの発振周波数の分散は 5.97% となった。FPGA の経年劣化に関する測定では温度を室温 (28 °C)、80 °C または 100 °C に保ち、10,000 秒経過までの発振周波数の変化を測定した。測定の結果、温度が高いほど発振周波数の劣化が大きく、高温では 10,000 秒経過時点で 0.1% 程度の劣化が観測出来た。

キーワード NBTI、FPGA、ばらつき、経年劣化

## Degradation of Oscillation Frequency of Ring Oscillators Placed on a 90 nm FPGA

Shouhei ISHII<sup>†</sup> and Kazutoshi KOBAYASHI<sup>†,††</sup>

<sup>†</sup> Graduate School of Science & Technology, Kyoto Institute of Technology  
<sup>††</sup> JST, CREST

**Abstract** We focus on issues related to degradation of FPGAs which has become dominant due to scaling and quantitatively estimate the degradation of FPGAs by NBTI. We map ring oscillators on the Cyclone II FPGAs and measure the variation of oscillation frequency. In the result, the variation of oscillation frequency is 5.97%. As for degradation of FPGAs, we measure the variation of oscillation frequency until 10,000 seconds passed at room temperature (28 degrees), 80 degrees or 100 degrees. As the result, degradation of oscillation frequency increases as temperature became higher and degradation of about 0.1% at 10,000 seconds was observed at high temperature.

**Key words** NBTI, FPGA, Variation, Degradation

### 1. 序 論

集積回路技術は飛躍的に進歩し、1 つのチップに搭載されるトランジスタの数は、ここ数十年で急激に増加した。その結果、素子が微細化して集積度が増し、動作も高速化した。一方、素子数が増えたことによる素子のばらつきの問題が顕在化してきた。素子が微細化することにより、ばらつきは増大し、経年劣化の影響も受けやすくなる [1]。

このように微細化が進むにつれて、様々な問題が発生しているが、本論文では特に FPGA (Field Programmable Gate Array) のばらつきと FPGA の NBTI (Negative Bias Temperature Instability) による経年劣化を実測により定量的に求めることを目的とする。NBTI とは、pMOS のゲートソース間に負バイアスが印加されることで、時間の経過とともにデバイスの閾値電圧が大きくなっていく現象である [2]。閾値電圧が大きくなることで、トランジスタの速度は遅くなる。劣化に関して、動的 (Dynamic) に劣化させた場合と、静的 (Static) に劣化させた場合で、NBTI による劣化の速度は異なっている [3]。

本論文では 2 節で FPGA のばらつきを求める方法について、3 節では FPGA の経年劣化を測定するための方法について述べる。4 節では 2 節、3 節で得られた結果を示し、それに対する考察を行い、5 節で結論を述べる。

2. FPGA のばらつきの測定

### 2. FPGA のばらつきの測定

ここでは、FPGA のばらつきを求めるために必要な RO (Ring Oscillator) を FPGA に実装する方法、発振器の配置を自動化する方法、発振周波数の測定を自動化する方法について述べる。

#### 2.1 発振回路の作成

発振回路と発振周波数を分周するための回路の構造を図 1 に示す。図 1 は RO と分周器 (DIV:Divider) と入出力ピンで構

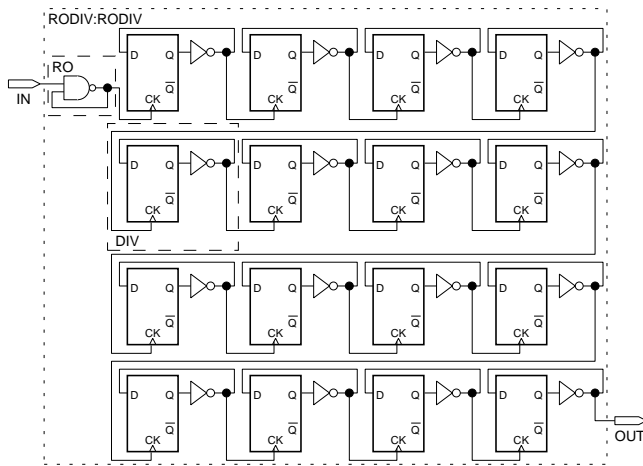


図 1 発振回路と発振周波数を分周する回路

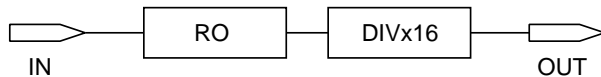


図 2 RO → DIVx16(DIV が 16 個) の回路図

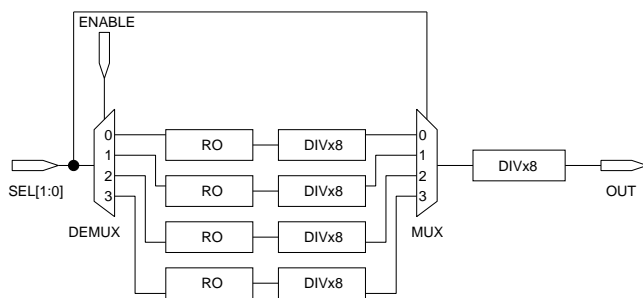


図 3 2 ビット入力切替器での DEMUX → RO → DIVx8 → MUX → DIVx8 の回路図

成され、発振させたものを分周器を用いて分周し、その発振周波数を調べる。分周させて周波数を落として測定することで、ノイズの影響の少ない正確な発振周波数を求めることが出来る。図 1 の簡略図を図 2 に示す。図 1 が今回測定を行う回路ではあるが、1 つの RO ではなく複数の RO をマッピングした方が一度にまとめて測定出来るため、図 3 のような DEMUX (Demultiplexer) と MUX (Multiplexer) を用いて、多数の発振器を切り替える。これにより、1 回の Configuration (FPGA に実装する回路構造データをダウンロードすること) で、多数の発振器の発振周波数が測定出来る。

図 3 中の Enable が High の時、RO が発振する。図 3 では、DEMUX を使うことにより、出力数を増やして多数の発振器の発振周波数が測定出来る。MUX は DEMUX によって複数出力されたものを 1 つにまとめるものである。セレクタを切り替えることで、多数の発振器の発振周波数を求めることが出来、マッピングする場所を変えて、発振周波数の値を比較することで、FPGA のばらつきを求めることが出来る。

ばらつきを一度にたくさん調べたい場合は、入力切替器のビット数を増やして一度に同時に測定出来る発振器の数を増やせばよい。しかし、入力のビットを増やすと回路の構造が大き

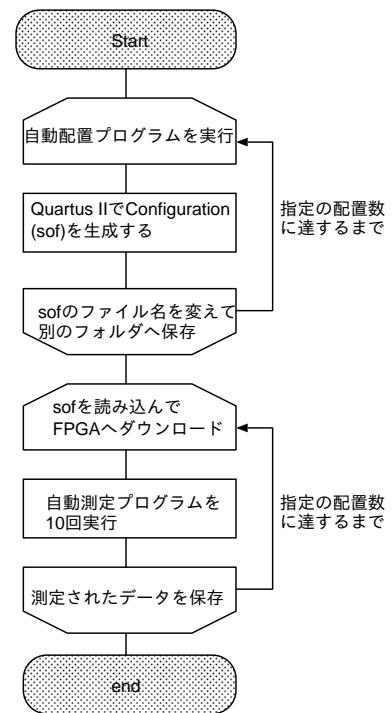


図 4 自動化のフローチャート

く変わってしまい、それが測定に悪い影響を及ぼす。そこで本研究では 2 ビットの入力切替器を用いて、4 つの発振器の発振周波数を調べ、その発振器の配置を自動で変えることで FPGA のばらつきを調べることにした。実際に多数の発振器を様々な場所にマッピングして得られた発振周波数の測定結果と考察については 4 節で述べる。

## 2.2 配置と測定の自動化プログラム作成

ここでは、FPGA のばらつきを求めるために必要な配置と測定の自動化について述べる。自動化を行うためのフローチャートを図 4 に示す。このような手順で、FPGA のばらつきを自動で求める。配置の自動化についての詳細は 2.2.1 節、測定の自動化についての詳細は 2.2.2 節に示す。

### 2.2.1 配置の自動化

発振器の配置する場所を指定することは出来、短時間で FPGA のばらつきを求めるために FPGA にマッピングを行う自動配置プログラムを作成した。発振器などの配置には Altera 社の Quartus II を用いており、これには配置配線を行った際に生成される qsf (Quartus II Settings File) の中身を変更することで、配置したい場所を自由に指定出来る。

qsf と Verilog あるいは VHDL で記述した回路で、FPGA に回路構造データをダウンロードするための sof (SRAM Object File) が生成可能である。本研究では qsf を Perl で自動的に書き換え、これをバッチ化して多数の qsf、sof を生成した。

### 2.2.2 測定の自動化

sof が自動的に生成出来るようになったので、測定の自動化も行う。これには GPIO (General Purpose Input/Output) と呼ばれる FT245RL (USB パラレル変換モジュール) を用いて FPGA を制御して、カウンタを使って周波数を測定し、出力された周波数の値を PC にまとめて自動的に取り込む。

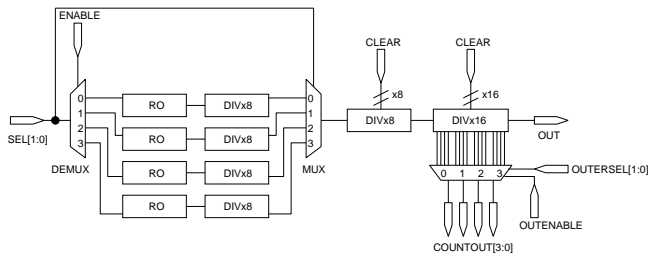


図 5 FT245RL で FPGA を制御するための回路図

それには図 3 の後段のカウンタ (分周器) を 24 段に増やす。これにより前段 (1~8 段目) の DIV を加えても 32 段あるため、 $2^{32} = 4,294,967,296 \approx 43$  億回程度カウント出来る。発振器から出力される発振周波数は 3GHz 程度であるから、1 秒間の発振時間だと、30 億回のカウントとなり桁溢れが起こらない。その後段 24 段のカウンタの出力を取り出して、FPGA から GPIO ピンへ送る。ピン数の関係上ここでは一番後ろ (最も分周しているところ) から 16 ビット分カウンタの出力を取り出した。取り込むカウンタの数を増やすことによって、発振周波数の精度が増し、その精度は 8 ビット分カウンタの出力を取り出す時 0.391%だが、16 ビットでは 0.00153%まで上がる。

16 ビット分のカウンタをまとめて取り出すにはピンが足りないので、セレクタを使って 4 ビットずつ取り出した。2 ビットの取り込むカウンタを切り替えるセレクタと 1 ビットの ENABLE を用いて、4 ビットずつ取り込む。これに元々あった RO を切り替える 2 ビットの入力切替器と 1 ビットの ENABLE を加え、新たに D-FF をリセットするために RESET を加えれば、GPIO ピンは  $7 + 2 + 1 + 1 = 11$  ピンで済む。D-FF にリセットを付けるのは、カウンタの値を出力した後に、リセットしないと前の値が D-FF に残ってしまうためである。

### 2.3 測定自動化のための回路

測定を自動で行うために、図 3 の回路にカウンタを増やして、新たに OUTERSEL、OUTENABLE、COUNTOUT、CLEAR を追加した回路図を図 5 に示す。入出力ピンに新たに OUTERSEL、OUTENABLE、COUNTOUT、CLEAR などを加えているが、これは FPGA から見てそれぞれ、カウンタを 4 ビットずつ取り込むための入力切替器、その入力切替器の ENABLE、出力されるカウンタ値、D-FF のリセットとなっている。GPIO ピンと FPGA の入出力ポートをジャンプワイヤで接続して、カウンタの値をやり取りする。

### 2.4 GPIO と FPGA との接続手順

GPIO と FPGA との接続の手順を説明する。FT245RL は 1 個で 8 ビットしか GPIO ピンを持たない。しかし、今回の測定では 11 ビットの GPIO ピンが必要となるため、これを 2 個用いた。

今回の測定で用いるのは GPIO Port (DB0~DB7) と GND、USB Port である。ブレッドボード上に FT245RL を配置して、8 ピンの GPIO Port を使って、FPGA ボードの入出力ポートとジャンプワイヤで接続する。USB ケーブルで FT245RL から PC へと接続することで PC から電源供給がされて、PC 上にカウンタの値を表示出来る。FT245RL と FPGA ボードは

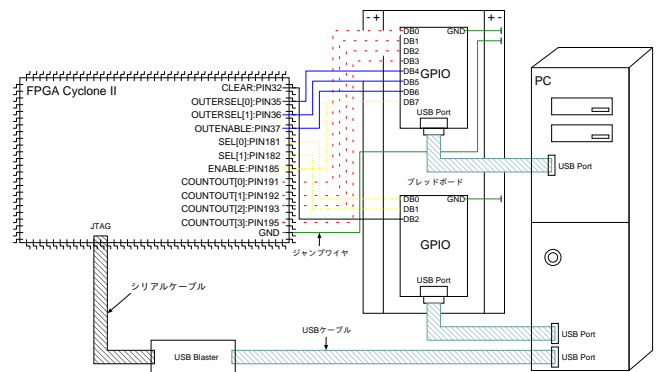


図 6 FPGA、GPIO、PC の概略図

それぞれ接地しておく。FPGA、GPIO、PC との概略図を図 6 に示す。FPGA のピンの役割はそれぞれ

- 発振器の切替器 (SEL[0]、SEL[1])
- セレクタ ENABLE ピン (ENABLE)
- D-FF のリセット (CLEAR)
- カウンタ切替器 (OUTERSEL[0]、OUTERSEL[1])
- セレクタ ENABLE ピン (OUTENABLE)
- カウンタ出力値 (COUNTOUT[0]~COUNTOUT[3])

である。

FPGA へのダウンロードは CUI コマンドで Configuration が出来るように、JTAG を使った。これには USB Blaster が必要となる。GPIO と FPGA の接続をした後、FPGA を GPIO で制御するためにプログラムを作成する。発振時間を 1 秒と設定すれば、桁溢れが起こらず 4 つの RO の発振周波数を求めることが出来る。短くても桁溢れは起こらないが、GPIO を制御する Windows の仕様上、割り込み時間の誤差が大きく、短い発振時間では発振周波数の値が不正確な値となる。

## 3. FPGA の NBTI による経年劣化の検証

この節では、FPGA の NBTI による経年劣化を調べるために、室温 (28 )、80、100 での経年変化による発振周波数の推移を求める方法について述べる。

FPGA 全体の経年劣化を調べるためには、ある特定の箇所に発振器を配置して一定期間発振周波数の推移を調べる。複数ヶ所における発振周波数の推移を調べなければ、FPGA 全体の経年劣化を調べたとは言えないので、その後別の場所に発振器を配置し、そこでも一定期間発振周波数の推移を調べる作業を繰り返す。複数ヶ所を自動で測定するバッチファイルを作成すれば、後は温度の調節だけ行えばよい。

NBTI は温度による影響が大きく、温度が高ければ高いほど、発振周波数の劣化も大きくなり、その劣化速度も増える。今回は温度による影響も確認するために、室温以外に、80、100 でも同様の測定を行った。発振を止めて DC ストレスを与えて発振周波数を劣化させる実験の測定結果と考察については 4 節で述べる。

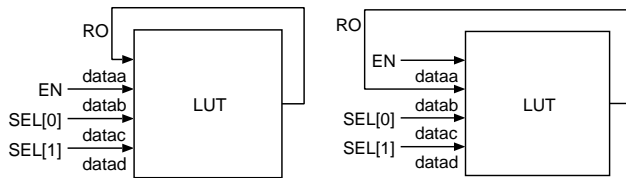


図 7 dataa 型

図 8 datab 型

#### 4. 結果と考察

この節では、2、3 節で述べた方法で測定を行い、それによって得られた結果とその結果から分かる考察を述べる。

##### 4.1 FPGA のばらつきの測定結果

2.2.1、2.2.2 節にある方法を用いて、FPGA のばらつきの測定を行った。測定に用いた回路は図 5 であり、測定の精度を高めるために、測定は同じ配置でも 10 回行った。測定にかかった時間は、図 4 における、配置配線と sof の作成が 1 回のループで 15 秒強、Configuration や同じ配置での 10 回の測定が 1 回のループで 45 秒であった。測定箇所は 112ヶ所行ったので、測定にかかる合計時間は  $112 \times (15 + 45)$  秒 = 112 分程度であった。

FPGA のばらつきを測定するには、そのばらつき以外に、測定誤差によるばらつき、LUT に入力される配線の位置の違いによるばらつきがあり、それらをなるべく取り除かねば正確な値は得られない。測定を行ってこれらのばらつきで一番大きいのは、LUT に入力される配線の位置の違いによるばらつきであった。図 7、8 に示すように、LUT に入力される配線が、それぞれ異なり、便宜上 dataa に RO が入る場合を dataa 型、datab に RO が入る場合を datab 型と呼んだ。datac や datad で発振しているものはなかった。他の入力 (EN、SEL[0]、SEL[1]) 同士の配線の位置に関しては、発振に全く影響しないので考える必要がない。発振周波数の測定値の範囲としては、測定した全ての場合において dataa 型が 1.3~1.8GHz、datab 型が 2.3~2.5GHz 程度の値となっていた。この結果から RO の発振経路によって大きく発振周波数の値が変わってしまうことが分かった。逆に、配線の位置をこちらで指定し、全て同じ構造にすれば、LUT に入力される配線の位置の違いによるばらつきの問題は解決出来るが、FPGA (Cyclone II) の仕様上配線は自動的に最適化されてしまい、こちらで変更する方法はなかった。測定結果では dataa 型のみで構成されるものは 1 つもなかったので、dataa 型が含まれるものを全て取り除いて、datab 型のみとし、LUT に入力される配線の位置の違いによるばらつきを取り除いた。

測定誤差によるばらつきについては、測定回数を増やせば増やす程、その平均値の精度が増すので、測定回数を 10 回に増やして測定誤差によるばらつきをなるべく小さくしている。測定毎の発振周波数のばらつきがどの程度あるのか調べるために平均を求めた後、分散も求めた。表 1 にはその値が示しており、全体の 10 回の分散の平均は最大でも 0.1%程度であった。

これら 2 つのばらつきについても考慮した後、FPGA のばら

表 1 RO の発振周波数の測定値

	全体の平均 [GHz]	全体の 10 回の分散の平均 [%]
RO[0]	2.392	0.0827
RO[1]	2.447	0.0765
RO[2]	2.427	0.0703
RO[3]	2.439	0.117
全 RO の平均 [GHz]		全 RO の分散 [%]
RO	2.426	5.97

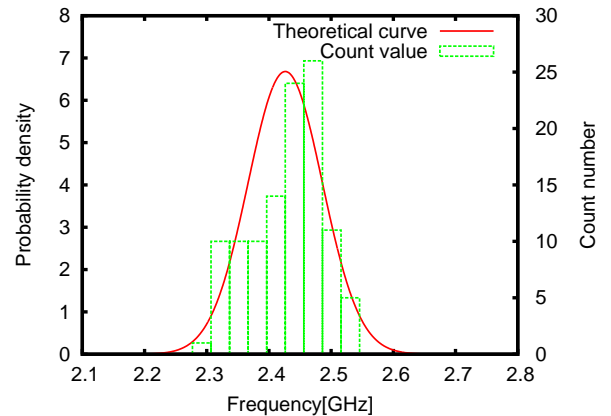


図 9 正規分布曲線と区間内の発振周波数のカウント数

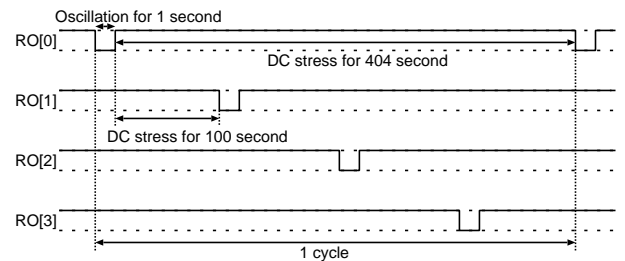


図 10 1 サイクルでのタイムチャート

つきが求められた。表 1 によれば、FPGA のばらつきを示す全 RO の分散の値は 5.97%となった。結果、FPGA のばらつきは測定毎のばらつきよりも 50 倍以上大きく測定毎のばらつきが FPGA のばらつきに与える影響は小さいことが分かる。

このばらつきが正規分布となっているかを確認するために、発振周波数を一定区間毎に区切り、その区間内にある発振周波数の個数をカウントした図を図 9 に示す。図 9 から、平均値に近づく程カウント数が多くなっているから、正規分布の形に近いことが分かる。

##### 4.2 FPGA の NBTI による経年劣化の測定結果

3 節にある方法を用いて、FPGA の NBTI による経年劣化の測定を行った。測定に用いた回路は図 5 である。1 サイクルの測定手順としては図 10 のタイムチャートのようにになっている。25 サイクル繰り返すと、 $400 \times 25 = 10,000$  秒 (3 時間弱) となる。これを合計 28 箇所で行って、RO の発振周波数が時間と共に劣化するかどうかを調べた。

室温、80、100 の条件下で、ある特定の配置場所における発振周波数の推移を図 11 に示す。図 11 から、100 での発振周波数が 80、室温よりも低下しており、100 では室温に比べ 30MHz 以上の低下が見られる。10,000 秒経過による劣

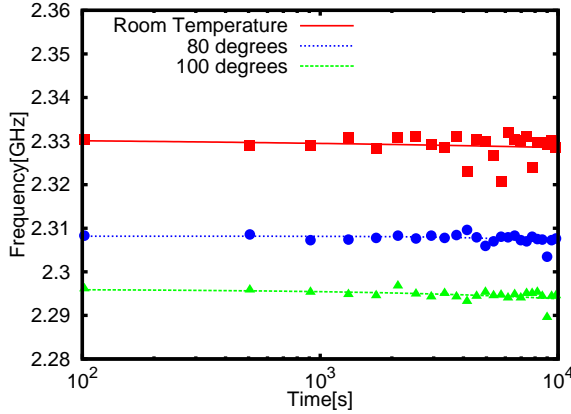


図 11 特定の配置場所における発振周波数の推移

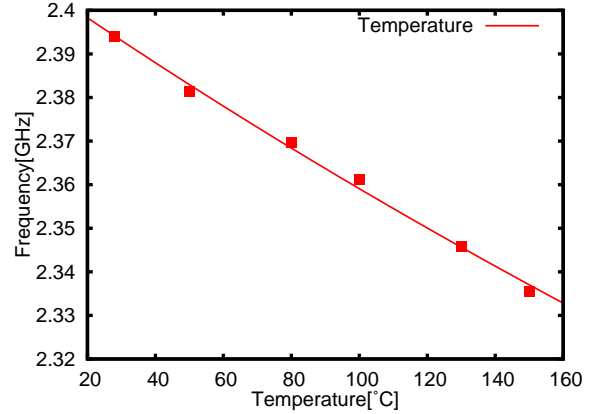


図 13 温度毎の発振周波数の変化

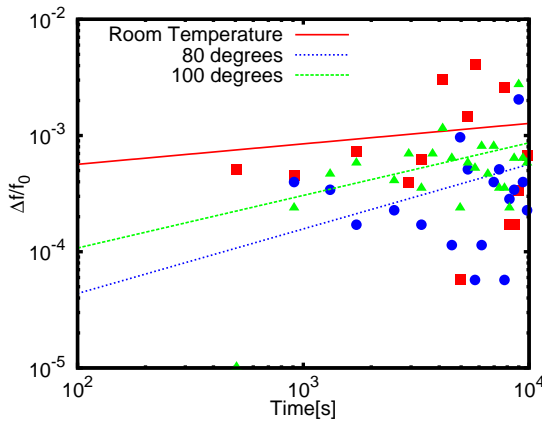


図 12 特定の配置場所における  $\frac{\Delta f}{f_0}$  の推移

化も僅かだが見られ、劣化が大きい箇所でも最大 0.1%程度の劣化が見られた。

しきい値電圧  $\Delta V_{th}$  は  $t^n$  に比例 [4] するので、

$$\Delta V_{th} \propto at^n \quad (1)$$

と書ける。発振周波数  $f$  としきい値電圧  $\Delta V_{th}$  も比例関係 [5] にあるので、 $f$  は

$$f = -f_0 at^n + f_0 \equiv at^n + f_0 \quad (2)$$

で表せられ、 $n$  が大きければ大きい程、劣化しやすい。発振周波数の経年劣化ではこの  $n$  の値は  $1/6 \sim 1/4$  という結果が多い [2], [6]。(2) 式を変形して  $f_0 - f$  を  $\Delta f$  とおいて

$$\frac{\Delta f}{f_0} \equiv \frac{f_0 - f}{f_0} = at^n \quad (3)$$

として  $t$  を横軸、 $\frac{\Delta f}{f_0}$  を縦軸とした経年劣化を表すグラフを図 12 に示す。この結果によれば、増減を繰り返しながらも徐々に  $\frac{\Delta f}{f_0}$  の値は大きくなっている。(3) 式によれば、時間  $t$  が経過するほど  $\frac{\Delta f}{f_0}$  の値が大きくなるので、図 12 は理想的な結果となった。 $\frac{\Delta f}{f_0}$  の増加量は配置場所や温度によって異なった結果となったが、その多くが 10,000 秒経過で、10~100 倍程度の増加となった。

各温度により、発振周波数がどの程度変化するかについても測定した。測定した温度は室温、50、80、100、130、

150 の 6 箇所である。測定回数は 1 つの温度で 5 回測定を行い、それらを平均して求めた。得られた結果のグラフを図 13 に示す。

図 13 から、温度が上がることによって発振周波数がほぼ線形に低下していくということが分かった。150 では室温の時と比べて、約 2.5%程度の発振周波数の低下が見られた。

#### 4.3 特定の箇所での FPGA の経年劣化の測定結果

1 箇所の測定が 10,000 秒では経年劣化が十分に確認出来なかったため、無作為に選んだ 1 箇所の Configuration ファイルを FPGA にダウンロードし、4.2 節と同じようにして発振周波数を求めた。ここでは発振を止めて DC ストレスを与える時間を 100 秒から 300 秒に、測定サイクルも 25 サイクルから 50 サイクルに増やして測定を行っている。得られた結果をそれぞれ図 14、15 に示す。図 12、15 から、どちらも増減を繰り返しながらも徐々に  $\frac{\Delta f}{f_0}$  の値が大きくなり、劣化していく様子が分かる。図 15 によれば増減のブレは  $10^4$  s を超えた辺りから徐々に小さくなっている。

経年劣化の進行も、図 11、14 を比べると、図 14 の方が、80 の場合  $10^4$  s 経過時点で、0.4%程度劣化しており、4.2 節の 0.1%の劣化よりも 4 倍大きい値となった。これは、発振を止めて DC ストレスを与える時間が 300 秒と増えているためである。以上の結果から正確な fitting 波形を得るには 30,000、40,000 秒程度の時間が必要で、経年劣化を十分に確認したい場合は DC ストレスを与える時間を 300 秒程度まで増やせば良い。

$t^n$  の  $n$  の値は、表 2 のように示された。1 place としているのが、特定の配置場所での  $a, n, f_0$  の値である。表 2 によれば、 $n$  はいずれもばらばらで、 $1/6 \sim 1/4$  の間にあるのは 1 箇所だけであった。この原因は、fitting した時に得られた  $a, n, f_0$  の分散があまりにも大きかったからである。fitting 波形から大きく外れた値を取り除いたところ分散は小さく出来たが、それでも大きく、実際に 1 place 80 degrees の  $n$  の値を変更し、0.25 としても fitting 波形にそれほど変化はなかった。このことから、短期間の測定 (1 日程度) では  $n$  が発振周波数の変動に与える影響は小さく、分散の値が大きくなり、分散の値の小さい  $n$  は求められなかった。



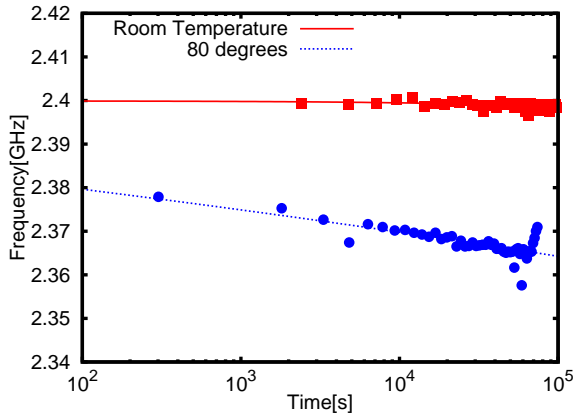


図 14 無作為に選んだ 1 箇所の発振器での発振周波数の推移

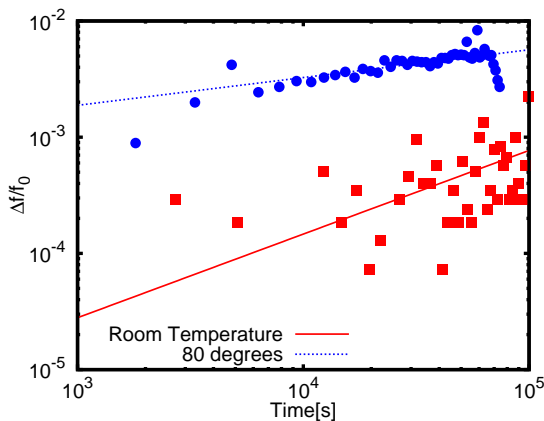


図 15 無作為に選んだ 1 箇所の発振器での  $\frac{\Delta f}{f_0}$  の推移

表 2 (2) 式の  $a, n, f_0$  の値

	$a$	$n$	$f_0$
Room Temperature	$-4.478 \times 10^{-4}$	0.194	2.331
80 degrees	$-6.588 \times 10^{-10}$	1.589	2.308
100 degrees	$-1.240 \times 10^{-5}$	0.5653	2.296
1 place Room Temperature	$-4.141 \times 10^{-6}$	0.5132	2.400
1 place 80 degrees	-0.05056	0.03356	2.439

## 5. 結 論

本稿では FPGA の NBTI による経年劣化の測定を行った。用いた FPGA デバイスは 90nm プロセスの Altera の Cyclone II である。ばらつきの測定では、配置と測定の自動化を行った後、発振器の発振周波数を測定した。実際に FPGA のばらつきの測定を行った際、ばらつきは FPGA のばらつき、測定誤差によるばらつき、LUT に入力される配線の位置の違いによるばらつきの 3 つがあった。

LUT に入力される配線の位置の違いによるばらつきに関しては、測定時に生成されるファイルを参照することで配線が LUT のどの位置に入っているかが分かるため、全て同じ配線パターンだけのものに絞れば、このばらつきは分離出来た。測定誤差によるばらつきは、測定回数を増やして平均を取ることで、なるべく測定誤差によるばらつきを小さくした。これらの影響をなるべく小さくし、そのばらつき (分散) を求めたところ、その

値は 5.97% となった。

FPGA の経年劣化の測定では、温度によって RO の発振周波数にどの程度影響を与えるか評価するため、温度を室温 (28 )、80、100 に保ち、10,000 秒経過までの発振周波数を求めた。発振周波数の経年劣化は最大でも 0.1% 程度で僅かであったが、28 箇所全てで、80、100 の方が室温よりも、0.5~1.5% 程度発振周波数は劣化していた。各温度 (室温、50、80、100、130、150) での発振周波数の値も温度が高い程ほぼ線形に発振周波数が低下していき、150 では室温と比べ 2.5% 程度劣化していた。結果としては、1 箇所における測定時間が短い点や、発振を止めて DC ストレスを与える時間が短かったこともあり、ほとんど経年劣化は見られず、3 時間で最大 0.1% 程度の劣化であった。

局所的な FPGA の経年劣化の測定では、DC ストレスを与える時間を 100 秒から 300 秒に増やし、測定時間においても 20 時間以上測った。その結果、80 の場合で 28 箇所測定した FPGA の経年劣化の測定と同じ 10,000 秒経過時点で、発振周波数の劣化量は 0.4% と 4 倍に増えていた。この結果から、FPGA の NBTI による経年劣化の測定をするためには、DC ストレスを与える時間が非常に重要になってくることが分かる。

今後の測定ではこの DC ストレスを与える時間を色々変えてみたり、より微細なプロセス (65、45nm) でも測定を行ってみたりして、それらが発振周波数の経年劣化にどの程度影響を与えるかを調べたいと考えている。

## 謝 辞

研究を進めるために様々な指導やアドバイスを下さった数内美智太郎さんには深く感謝致します。FPGA の経年劣化を測定するために、恒温装置を貸して頂いた京大の方々、様々なご協力を頂いた小林研究室の皆様にも感謝しています。

## 文 献

- [1] A. W. Strong, E. Y. Wu, R.-P. Vollertsen, J. S. G. LaRosa, I. Stewart E. Rauch and T. D. Sullivan: "Reliability Wearout Mechanisms in Advanced CMOS Technologies", A JOHN WILEY & SONS, INC. (2009).
- [2] K. Kang, K. Kim, A. E. Islam, M. A. Alam and K. Roy: "Characterization and estimation of circuit reliability degradation under NBTI using on-line IDDQ measurement", Proceedings of the 44th annual Design Automation Conference, DAC '07, New York, NY, USA, ACM, pp. 358-363 (2007).
- [3] W. Wang, S. Yang, S. Bhardwaj, S. Vruthula, F. Liu and Y. Cao: "The Impact of NBTI Effect on Combinational Circuit: Modeling, Simulation, and Analysis", Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, **18**, 2, pp. 173-183 (2010).
- [4] E. A. Stott, J. S. Wong, N. P. Sedcole and P. Y. K. Cheung: "Degradation in FPGAs: Measurement and Modelling", FPGA (Eds. by P. Y. K. Cheung and J. Wawrzynnek), ACM, pp. 229-238 (2010).
- [5] 松波 弘之, 吉本 昌広: "半導体デバイス", 共立出版株式会社 (2000).
- [6] K. Ramakrishnan, S. Suresh, N. Vijaykrishnan and M. Irwin: "Impact of NBTI on FPGAs", VLSI Design, 2007. Held jointly with 6th International Conference on Embedded Systems., 20th International Conference on, pp. 717-722 (2007).