

DA変換器のVCOを用いた自己校正技術の検討

平成25年2月28日

荒川 雄太, 小林 春夫, 松浦 達治, 元澤 篤史(群馬大学)

小林 修(STARC)

新津 葵一(名古屋大学)

OUTLINE

- 研究背景
- 研究目的
- アプローチ
- まとめ

OUTLINE

- 研究背景
- 研究目的
- アプローチ
- まとめ

研究背景

LSIの微細化

デジタル：高精細化、低消費電力化、高速の恩恵

アナログ：素子ばらつき大

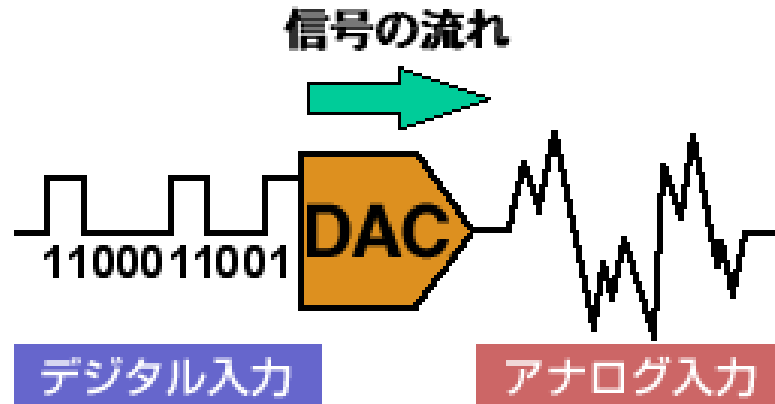


デジタル技術でアナログ性能向上

電流型DA変換器で検証！

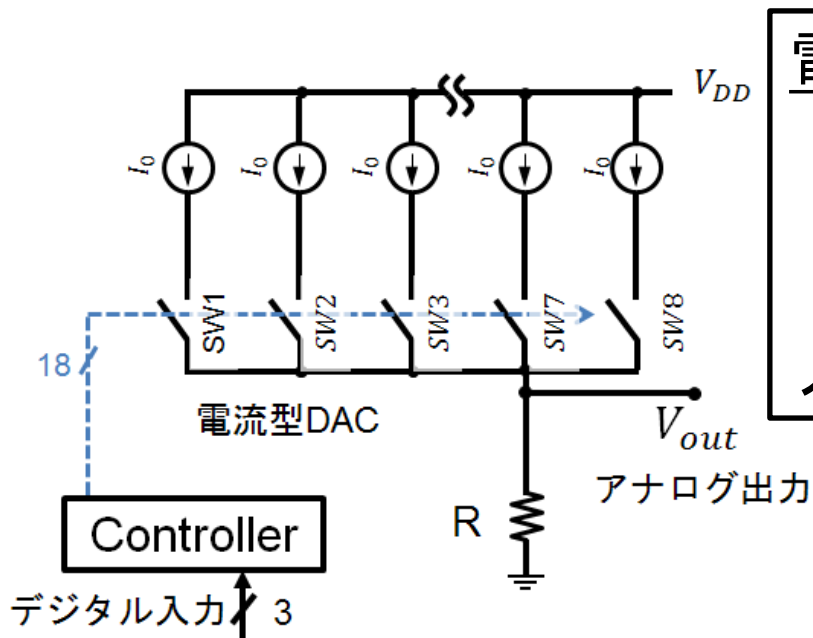
※デジタル-アナログ変換器(Digital-to-Anaalog Converter))

DA変換器



離散的なデジタル値
連続的なアナログ信号に変換

応用 携帯電話
MP3プレイヤー、地デジ放送など



電流型DA変換器

デジタル入力データに応じ
電流源をスイッチ

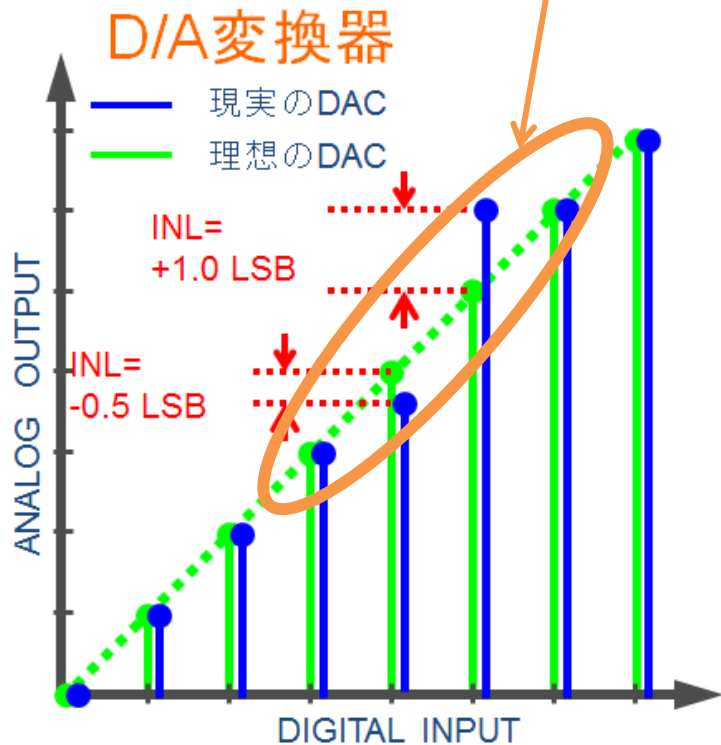
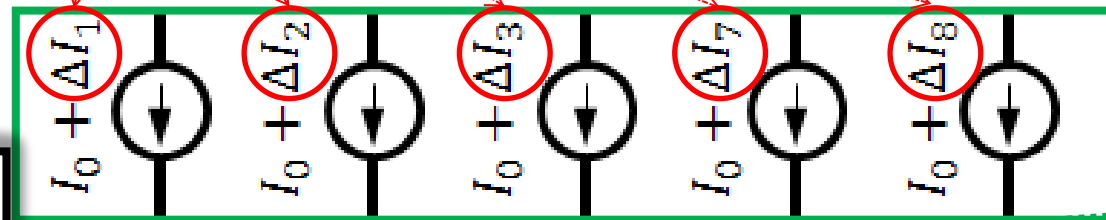
↓

入りに比例したアナログ電圧を出力

電流型DA変換器の問題点

各電流源間の電流量に**ばらつき**

DA変換器の**非線形性**



V_{DD}
 out
 ログ出力

OUTLINE

- 研究背景
- 研究目的
- アプローチ
- まとめ

研究目的

電流型DA変換器



デジタル技術で線形性向上

VCOで各電流源の大小を判定



各電流源を並び替えてDA変換器の線形性向上

VCOには線形性不要、単調増加特性のみでよい

電圧制御発振器 (Voltage-Controlled Oscillator: VCO)

OUTLINE

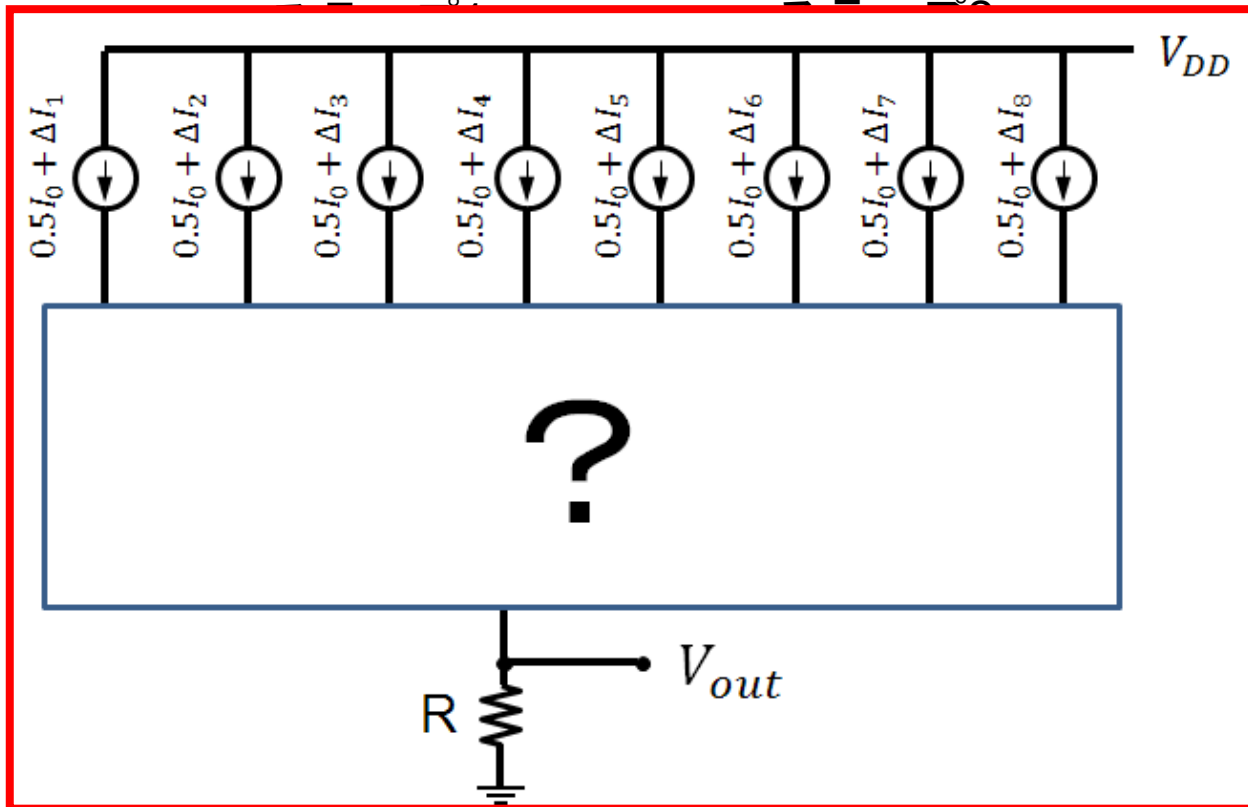
- 研究背景
- 研究目的
- **アプローチ**
- まとめ

電流源の並び替え

2bit DA変換器の場合

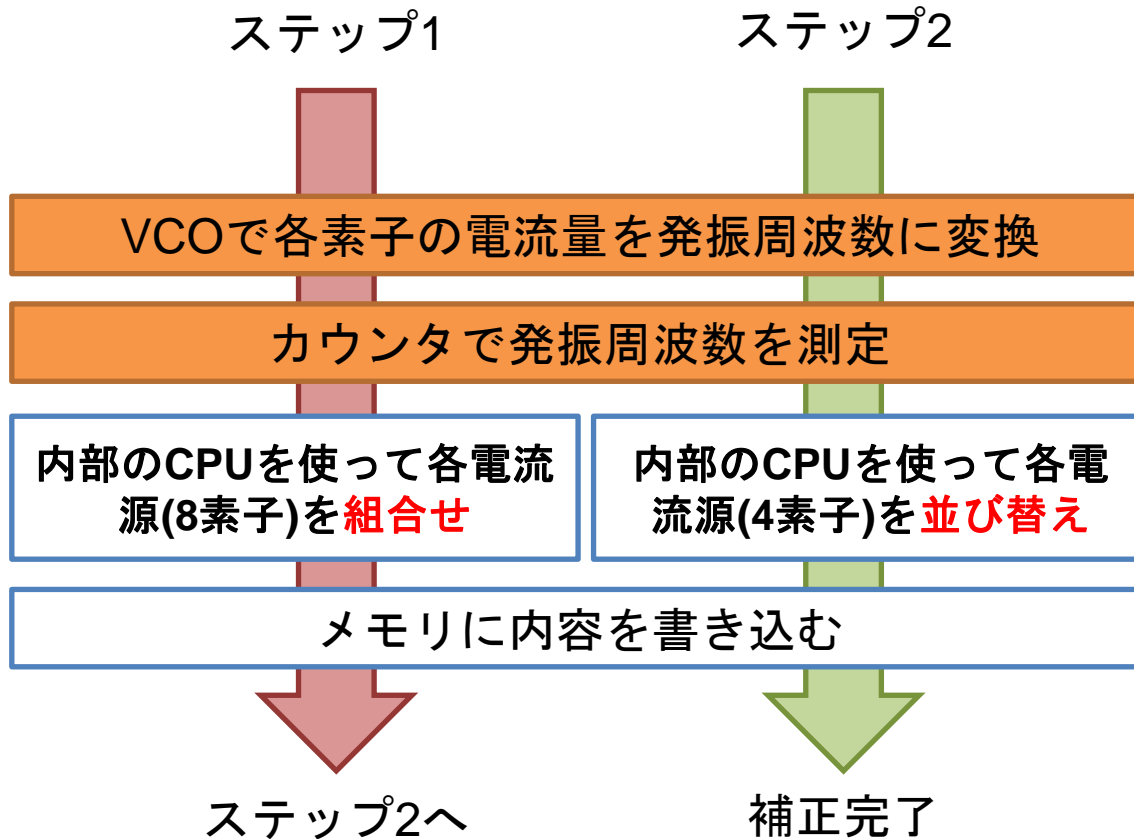
(1) $0.5I_0 + \Delta I_k$ の電流源を8個用意

8個を4個にする場合

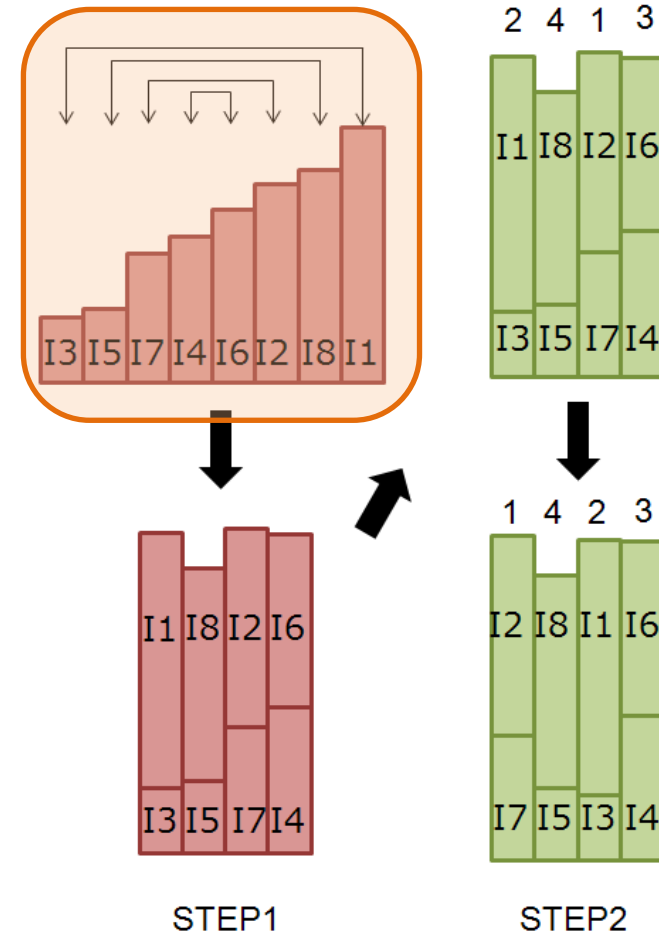


電流源の並び替え

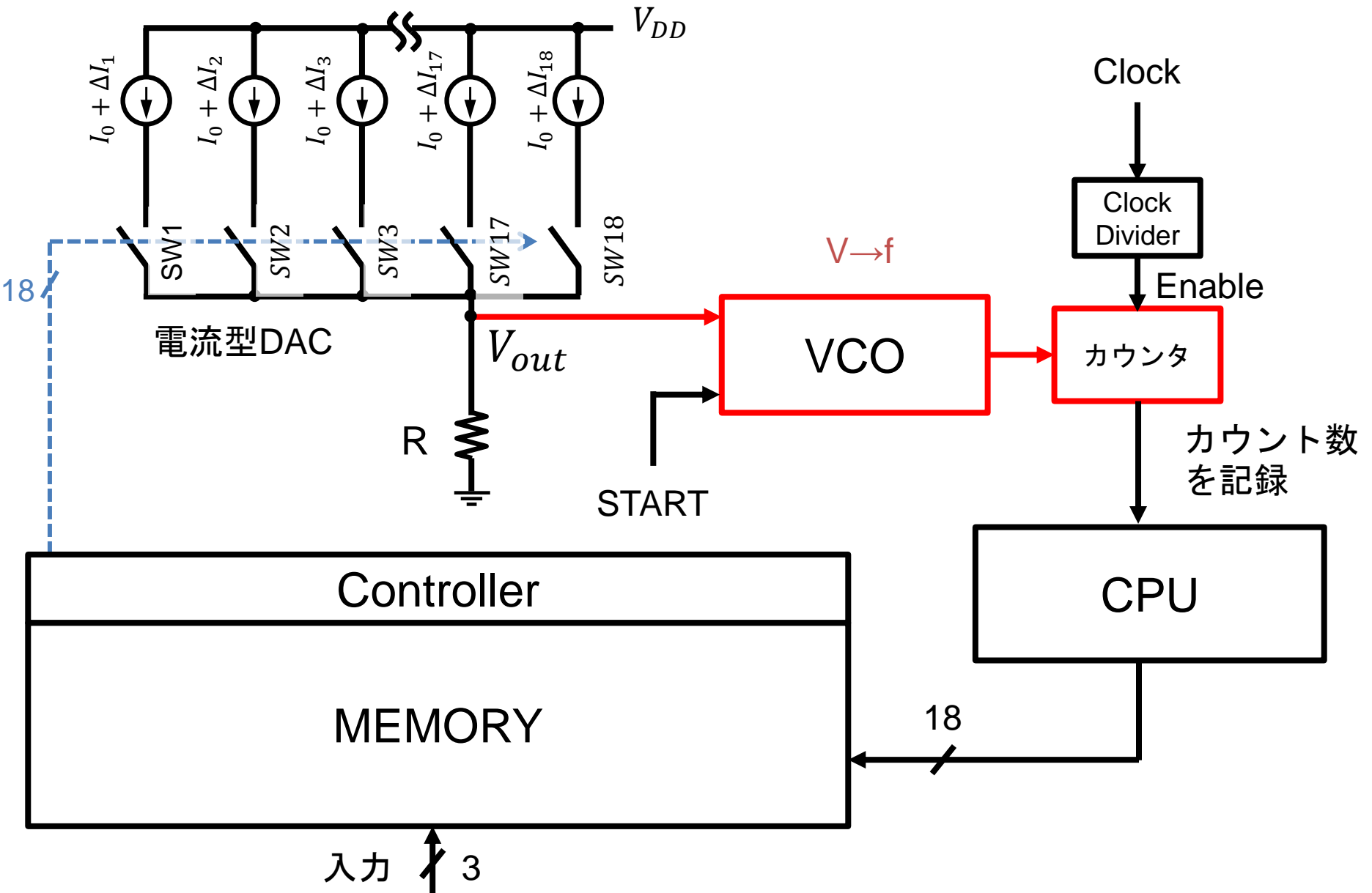
(2)VCOとカウンタで大きさ順番を計測



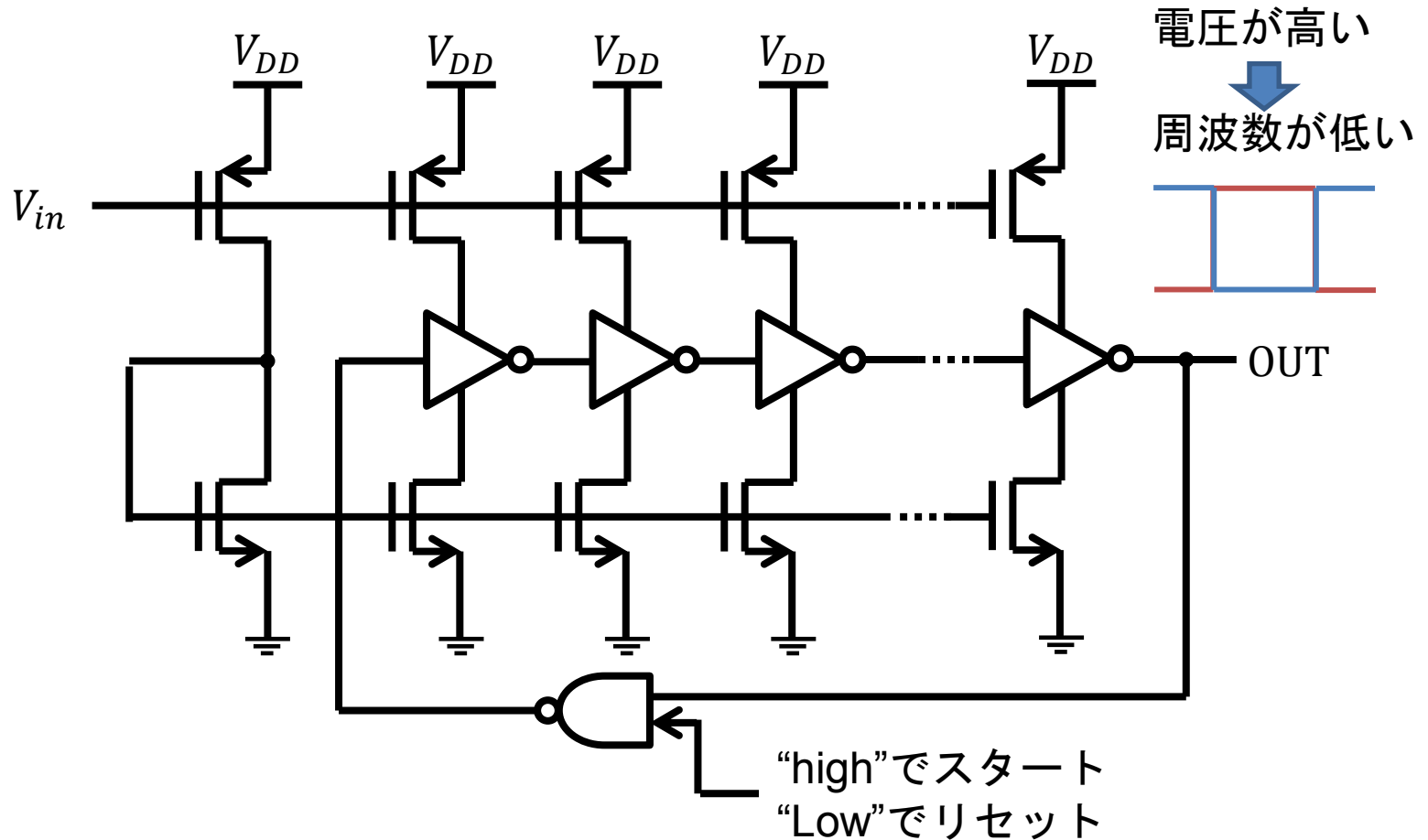
8個を4個にする場合



全体図

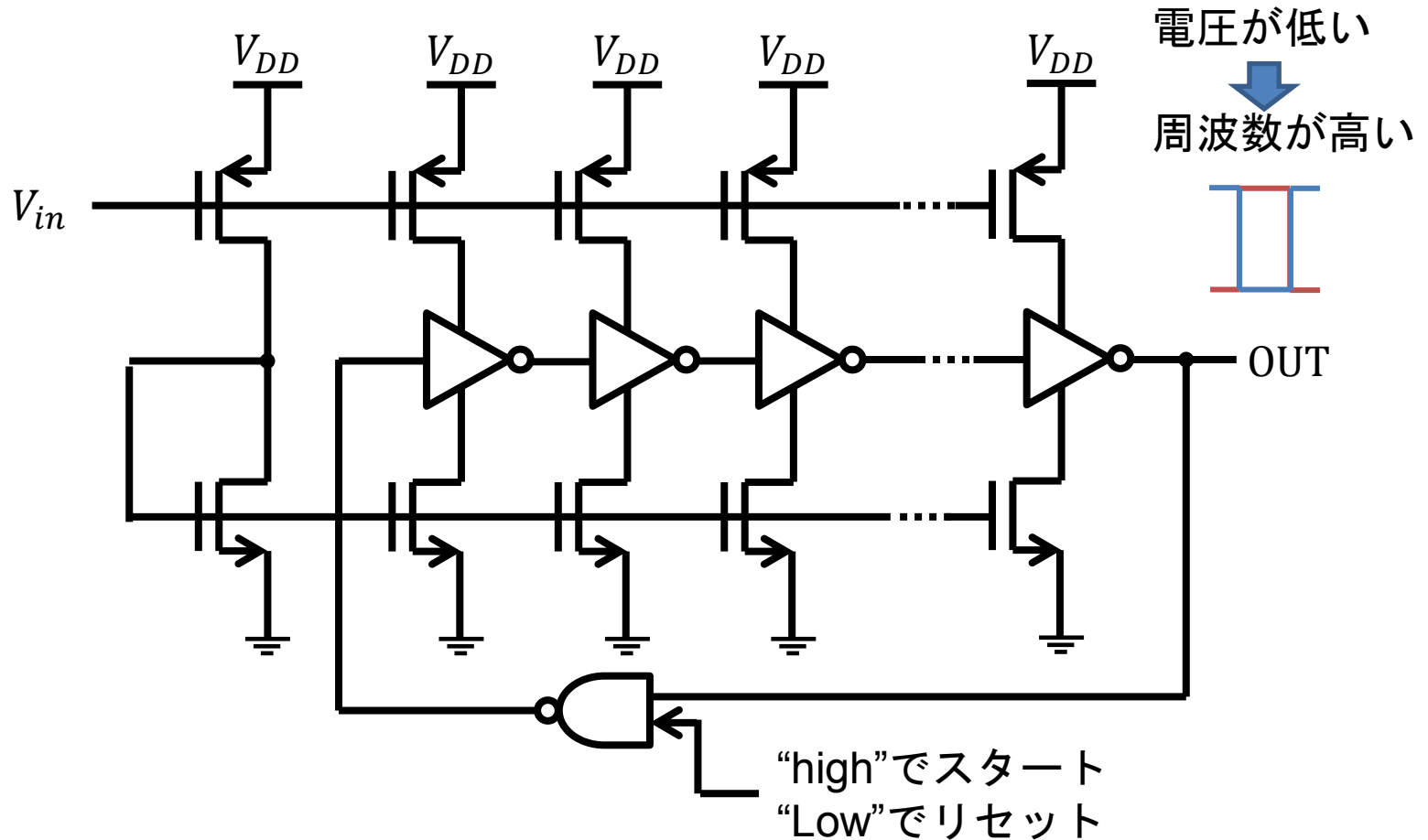


VCOの構成



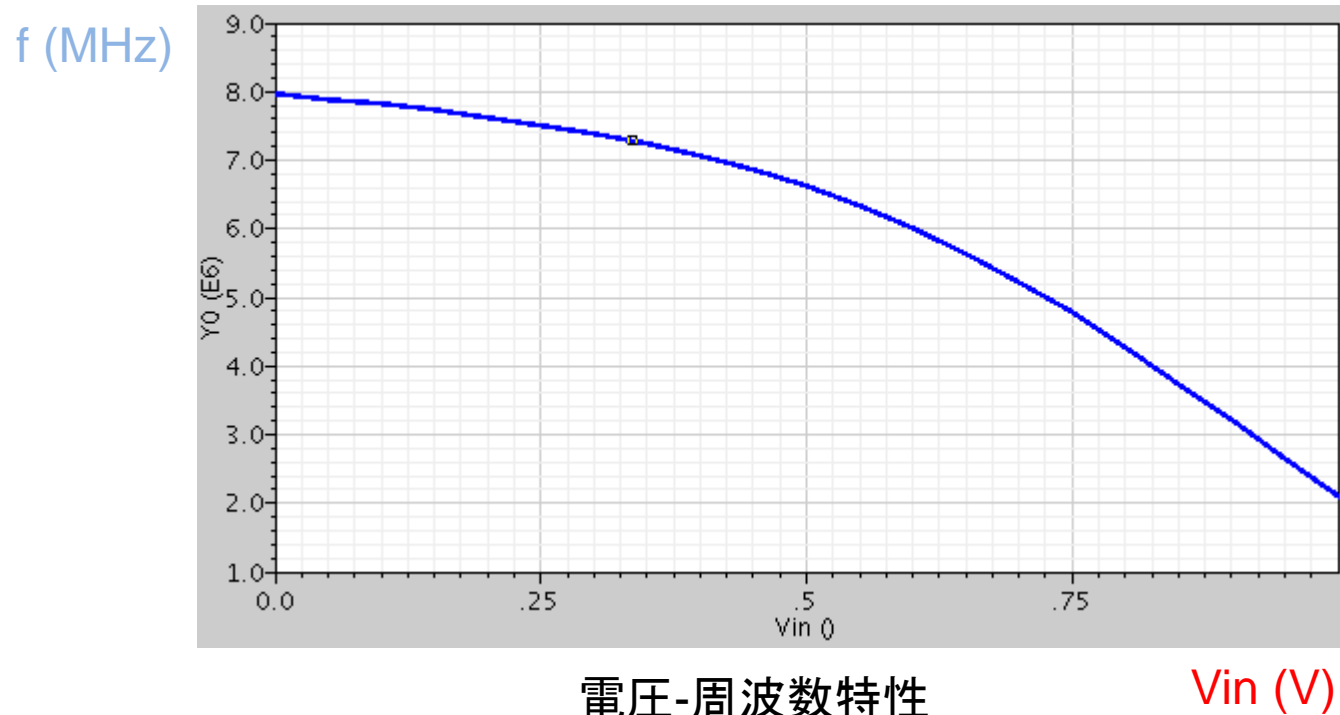
電圧 V_{in} を周波数 f に変換する回路!

VCOの構成



電圧 V_{in} を周波数 f に変換する回路!

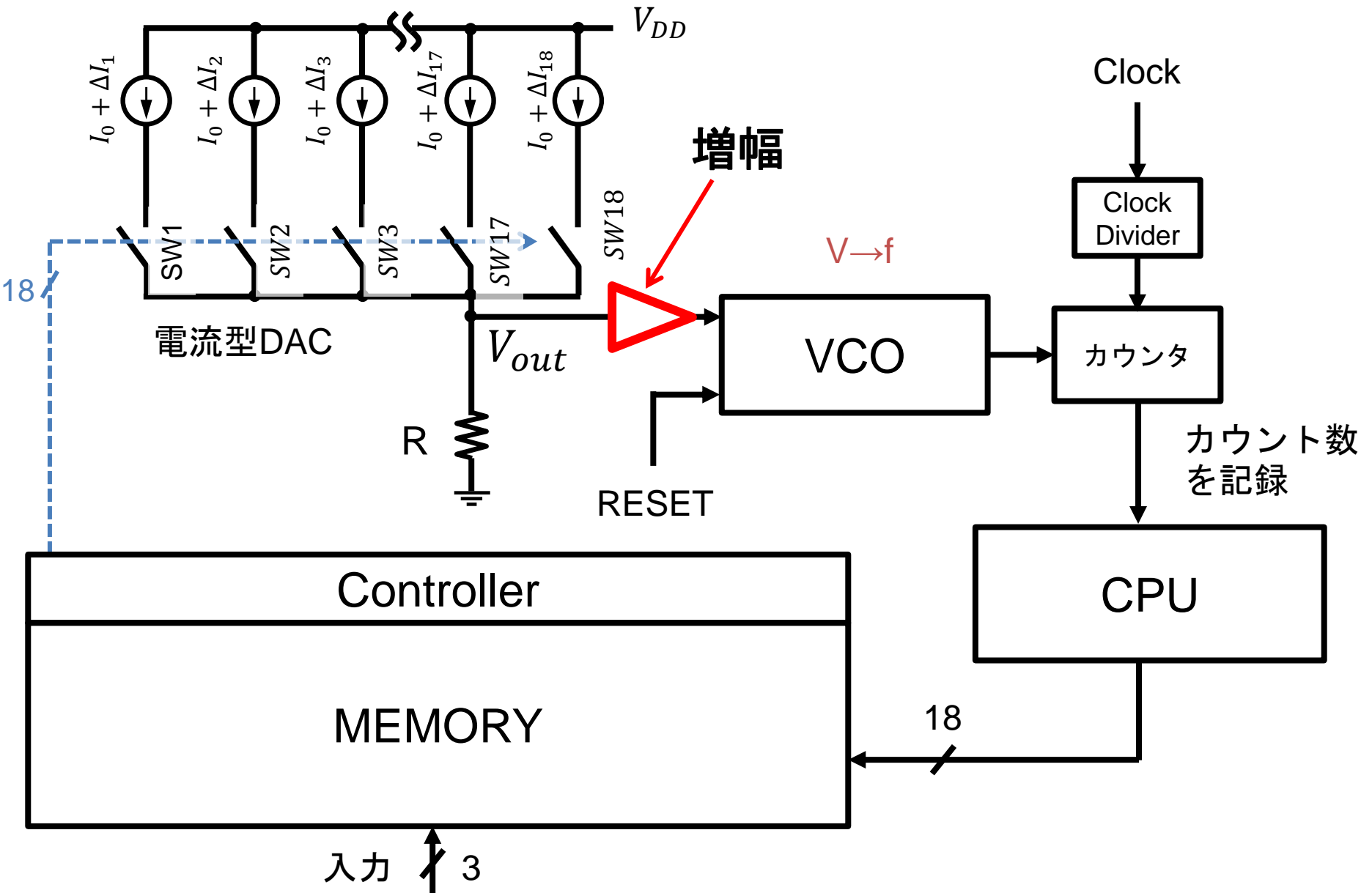
VCOの入力電圧による周波数の変化



単調増加性：入力電圧 V_{in} が大 ➡ 出力周波数 f が低

電流源の大小判定のため
単調減少性があれば線形性は不要
➡ 回路が簡単でよい

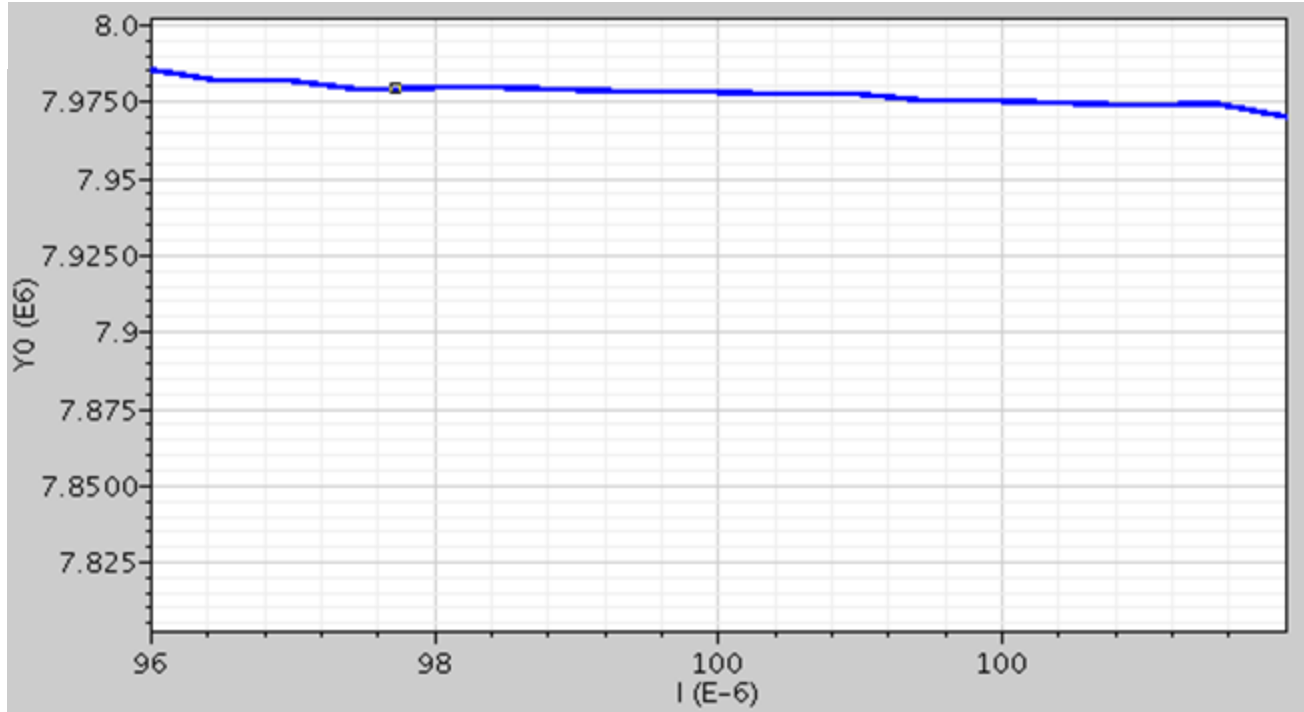
増幅器を設けて感度向上



電流源の電流値に対する発振周波数(利得1)

電流源の電流値 I :96~104 μ Aまで変化させると、

f (MHz)



I (μ A)

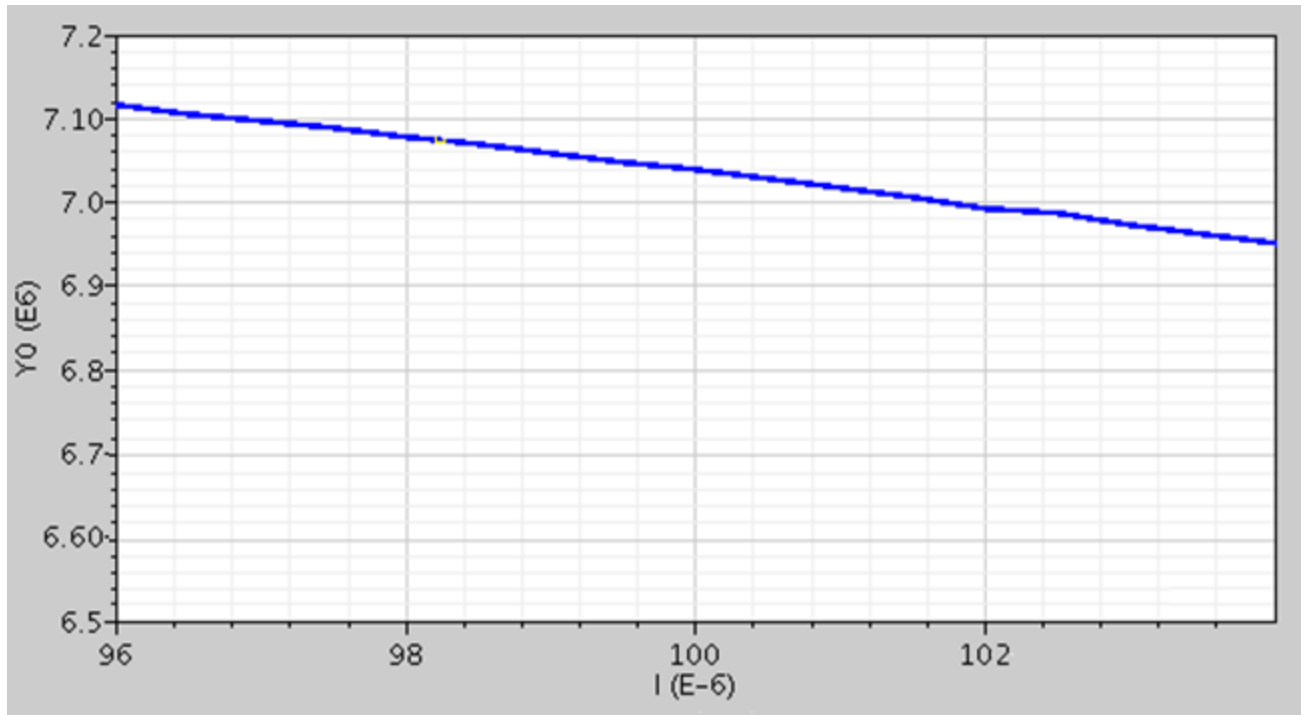
利得 1

出力周波数があまり変化せず、正確に測るには長い測定時間、回数が必要

電流源の電流値に対する発振周波数(利得5)

電流源の電流値 I :96~104 μ Aまで変化させると、

f (MHz)

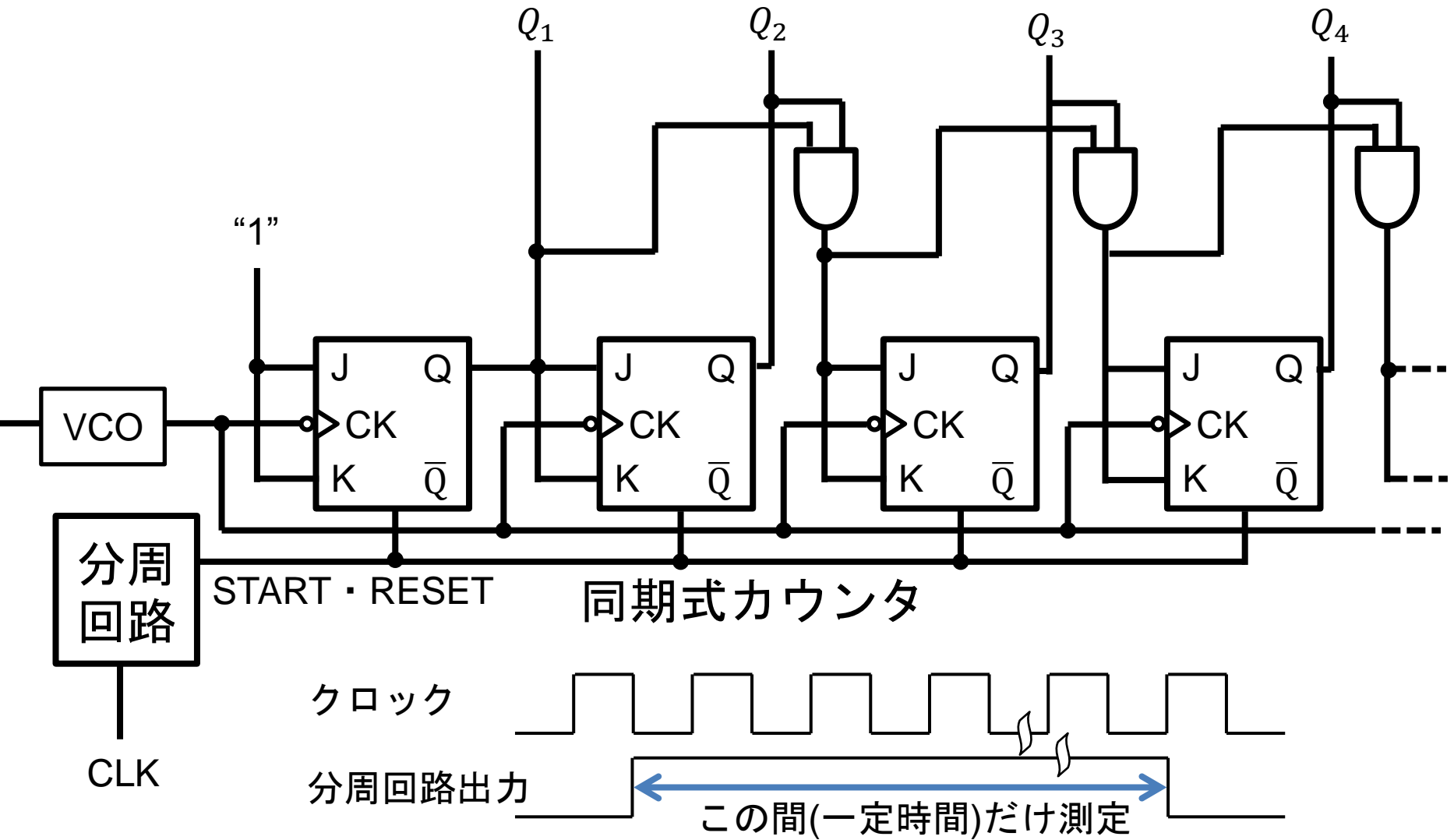


I (μ A)

利得 5に増幅

ばらつきによる影響を増幅→測定時間の低減

カウンタ回路

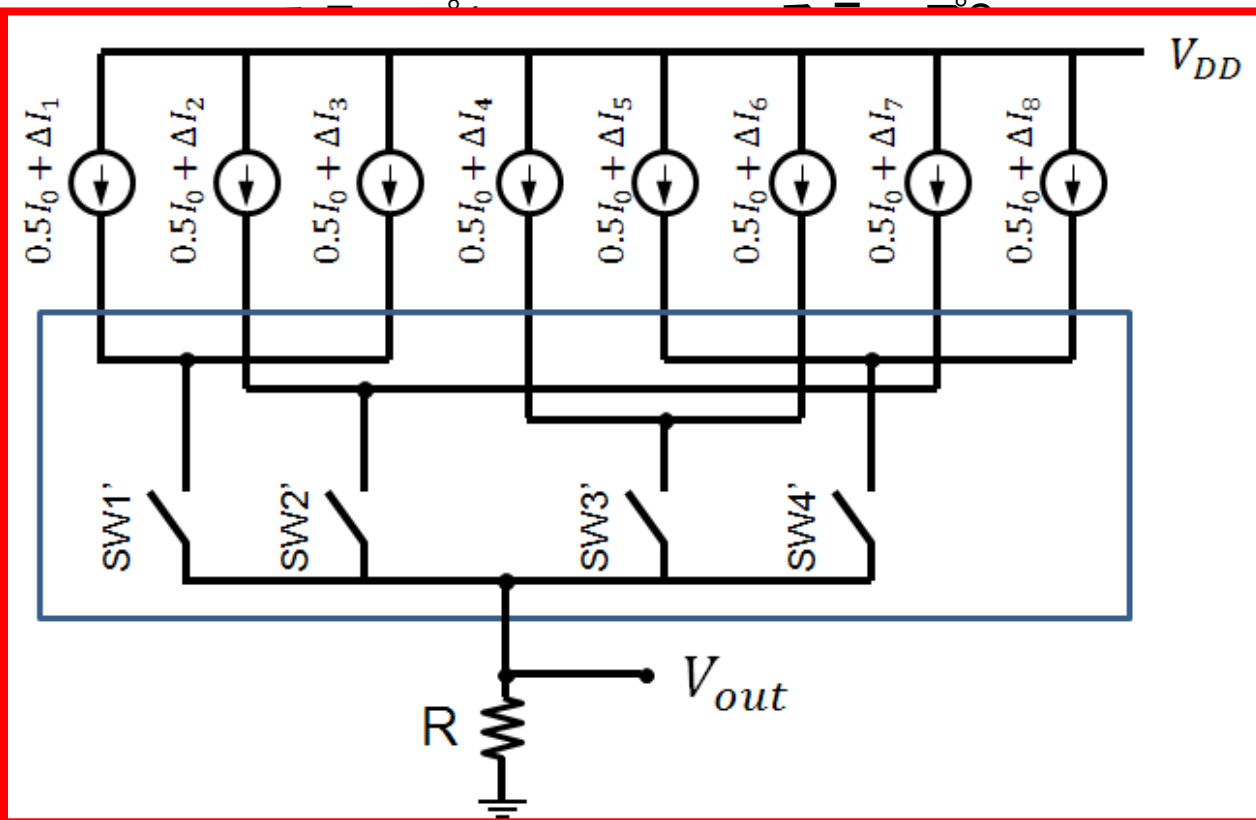


電流源の並び替え

(3) $0.5I_0 + \Delta I_i$ と $0.5I_0 + \Delta I_j$ を接続

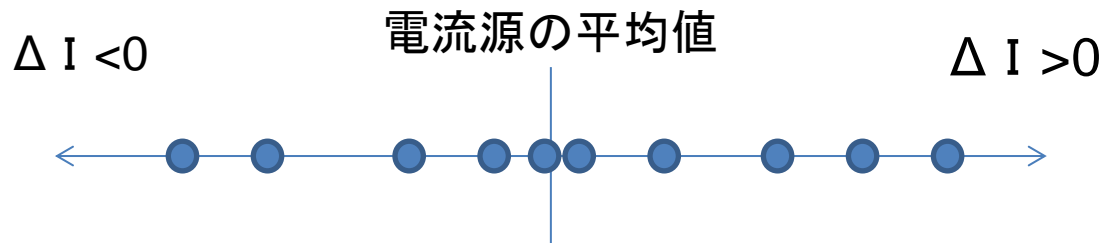
4個の $I + \Delta I_a$ を得る

8個を4個にする場合

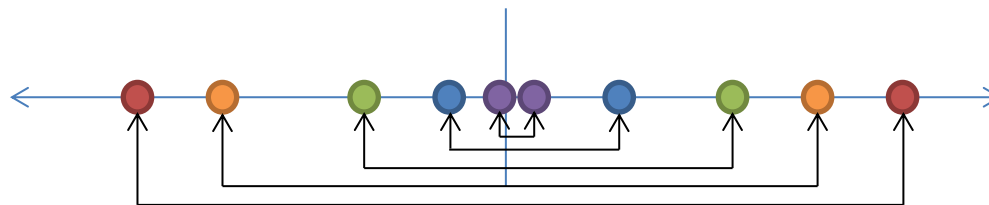


ばらつきを相殺する仕組み

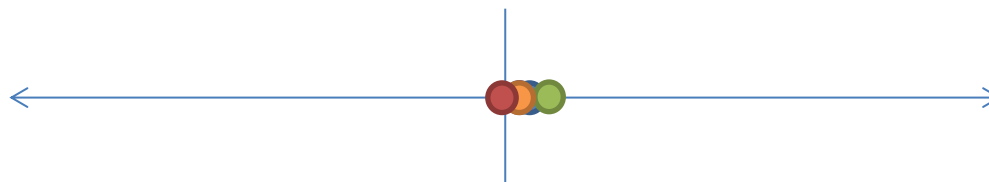
- を電流源の電流値



同色同士足し合わす



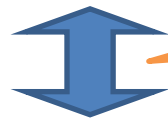
ばらつきが相殺される



数値シミュレーションでの乱数発生(3bit)^{22/38}

- 正規分布に基づく乱数発生を数十パターン検証

16 or 18 個の電流源を用意し、合成した場合



ばらつき度合(標準偏差 σ)を比較

単純に8個の電流源を設けた場合

※正規分布: $f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$

結果 16素子→8素子(3bit)

以下のようにシミュレーションを行った。

No.	正規乱数	順位	大きい順	並び替え合成後	ランダムな並び替え
1	98.41952071	10	111.351809	1位+16位	198.5665
2	100.1470139	8	108.14732	2位+15位	201.9763
3	106.2862442	3	106.286244	3位+14位	183.0993
4	95.69004684	12	105.550059	4位+13位	199.2607
5	91.91538053	14	102.014563	5位+12位	203.3289
6	91.1839458	15	101.618705	6位+11位	195.2033
7	97.64198894	11	101.31435	7位+10位	196.2434
8	101.6187051	6	100.147014	8位+9位	219.4991
9	102.0145631	5	99.7389809		
10	101.3143501	7	98.4195207		
11	99.73898087	9	97.6419889		
12	95.46428789	13	95.6900468		
13	90.69336774	16	95.4642879		
14	105.5500594	4	91.9153805		
15	111.3518094	1	91.1839458		
16	108.1473201	2	90.6933677		
AVE	99.82359904			199.6472	199.6472
STD	5.865493804			1.313224	9.483949

ばらつき
が相殺さ
れている

ランダム
な組み合
わせは更
にばらつ
く

大きい順に並び替え

分散が減っている

平均を100とし、標準偏差 $\sigma=7$ とした場合の乱数

これを数十から百パターン取る。

結果 18素子→8素子(3bit)

以下のようにシミュレーションを行った。

No.	正規乱数	順位	大きい順	並び替え合成後	ランダムな並び替え
1	96.02130891	13	111.67863	2位+17位	211.1677
2	111.6758633	10	111.232649	3位+16位	215.7048
3	99.49179101	10	106.236391	4位+15位	197.0439
4	104.4721989	5	104.58237	5位+14位	204.2788
5	111.2326493	2	104.472199	6位+13位	179.8637
6	103.6794142	6	103.679414	7位+12位	205.5304
7	93.3645182	16	101.730817	8位+11位	192.9872
8	104.5823697	4	100.8655	9位+10位	199.1882
9	99.6964059	9	99.6964059		
10	92.19905869	17	99.491791		
11	87.66465286	18	99.2940575		
12	106.236391	3	98.2855602		
13	99.29405748	11	96.0213089		
14	98.28556022	12	94.7016372		
15	94.7016372	14	94.0458541		
16	100.8655004	8	93.3645182		
17	94.04585407	15	92.1990587		
18	101.7308171	7	87.6646529		
AVE	99.95778047			AVE 199.9874	200.186
STD	6.235329209			STD 1.381723	10.66102

ばらつきが相殺されている

ランダムな組み合わせは更にばらつく

分散が減っている

平均を100とし、標準偏差 $\sigma=7$ とした場合の乱数

※18個の素子を用意し最も正・負の方向にばらつきのある素子は不使用

これを数十から百パターン取る。

16素子と18素子の結果比較(3bit)

- それぞれ100パターン取り、この分散(σ)の平均は、

素子数	並び替えた場合	ランダムな組合せ
16→8	2~2.4	8.8~9.3
18→8	1.7~1.8	8.8~9.3

テクニック不使用時 $\sigma=5$

※電流源の合成後を200になり、並び替えを不使用で電流源を用意すると標準偏差 σ は100の時の1/1.4になる。(7*1/1.4=5)

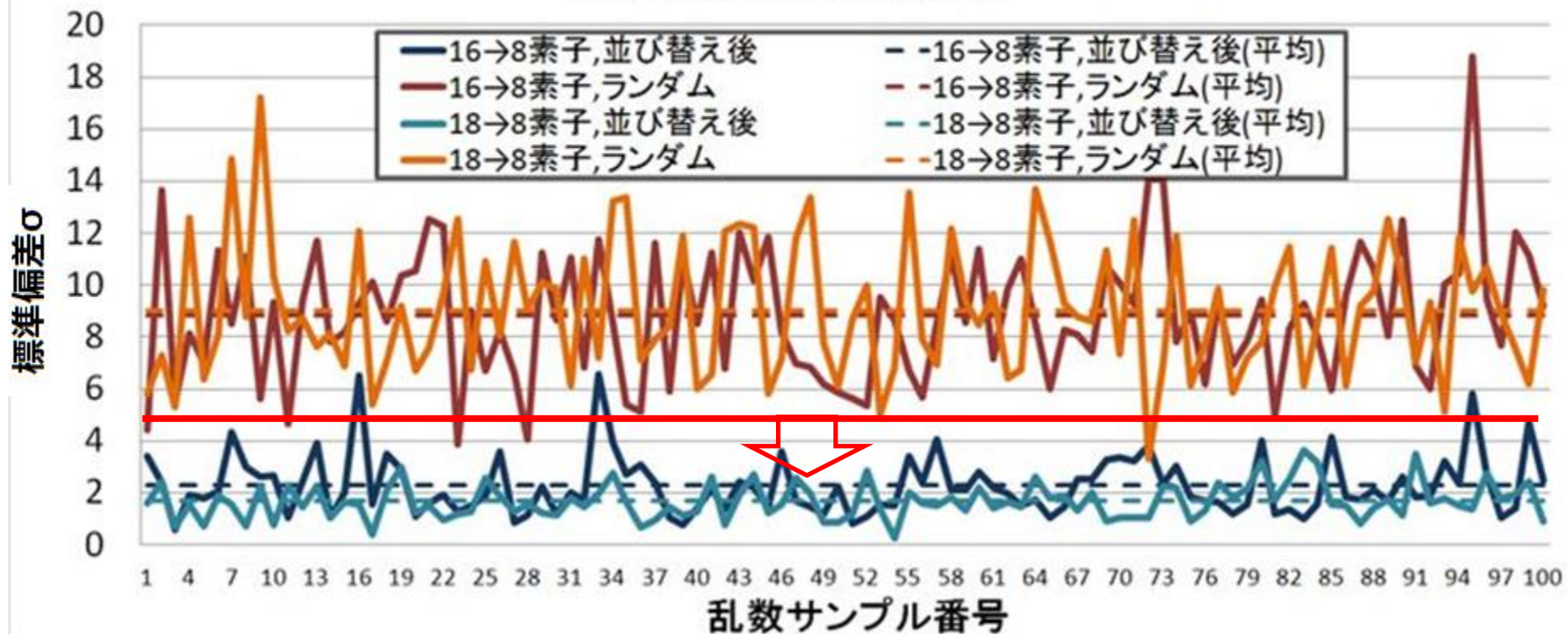
数値シミュレーション上の結果では50%~70%のばらつき(標準偏差)の低減を実現

分散係数の比較(100パターン)

素子数	並び替えた場合	ランダムな組合せ
16→8	2~2.4	8.8~9.3
18→8	1.7~1.8	8.8~9.3

テクニック不使用時 $\sigma=5$

並び替え合成後(200を理想とする)



電流源の並び替え

(4)メモリに組合せを書込み

ステップ1

VCOで各素子の電流量を発振周波数に変換

カウンタで発振周波数を測定

内部のCPUを使って各電流源(8素子)を**組合せ**

メモリに内容を書き込む

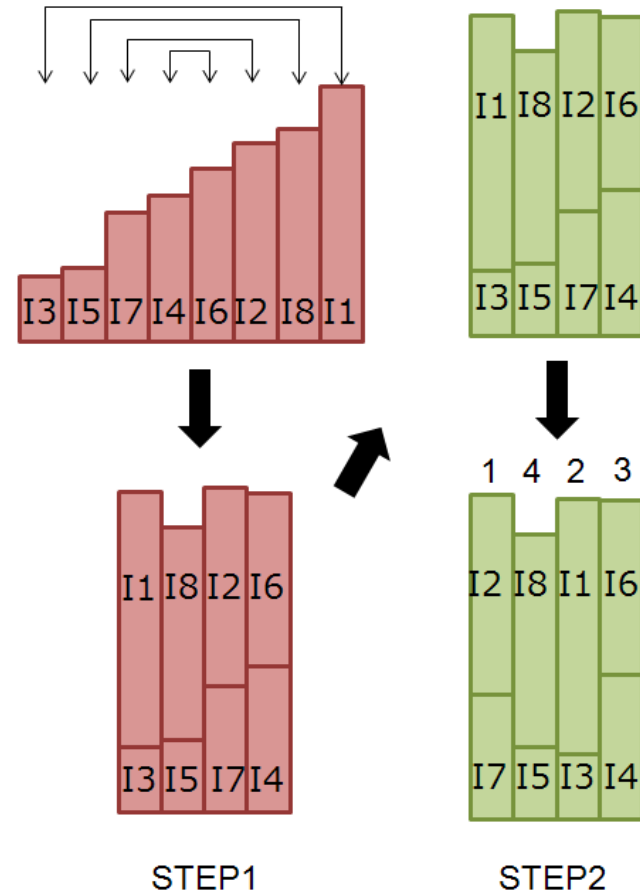
ステップ2へ

ステップ2

内部のCPUを使って各電流源(4素子)を**並び替え**

補正完了

8個を4個にする場合



キャリブレーションフロー

(5) $I + \Delta i_a$ をVCOとカウンタで大きさ
順番を計測

ステップ1

VCOで各素子の電流量を発振周波数に変換

カウンタで発振周波数を測定

内部のCPUを使って各電流
源(8素子)を**組合せ**

メモリに内容を書き込む

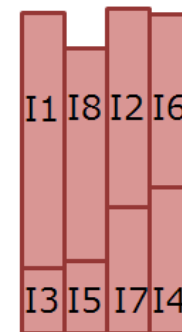
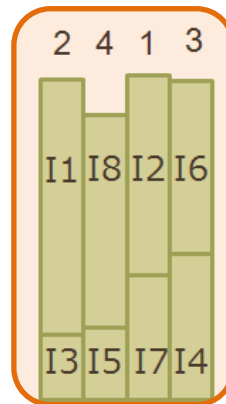
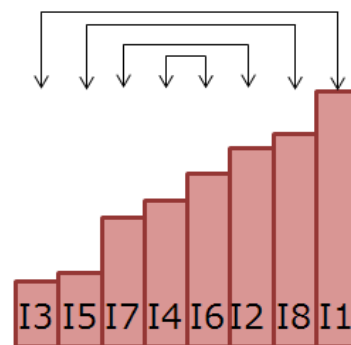
ステップ2へ

ステップ2

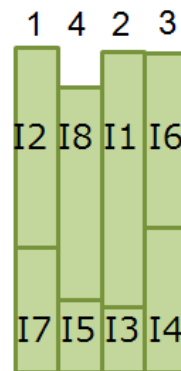
内部のCPUを使って各電
流源(4素子)を**並び替え**

補正完了

8個を4個にする場合

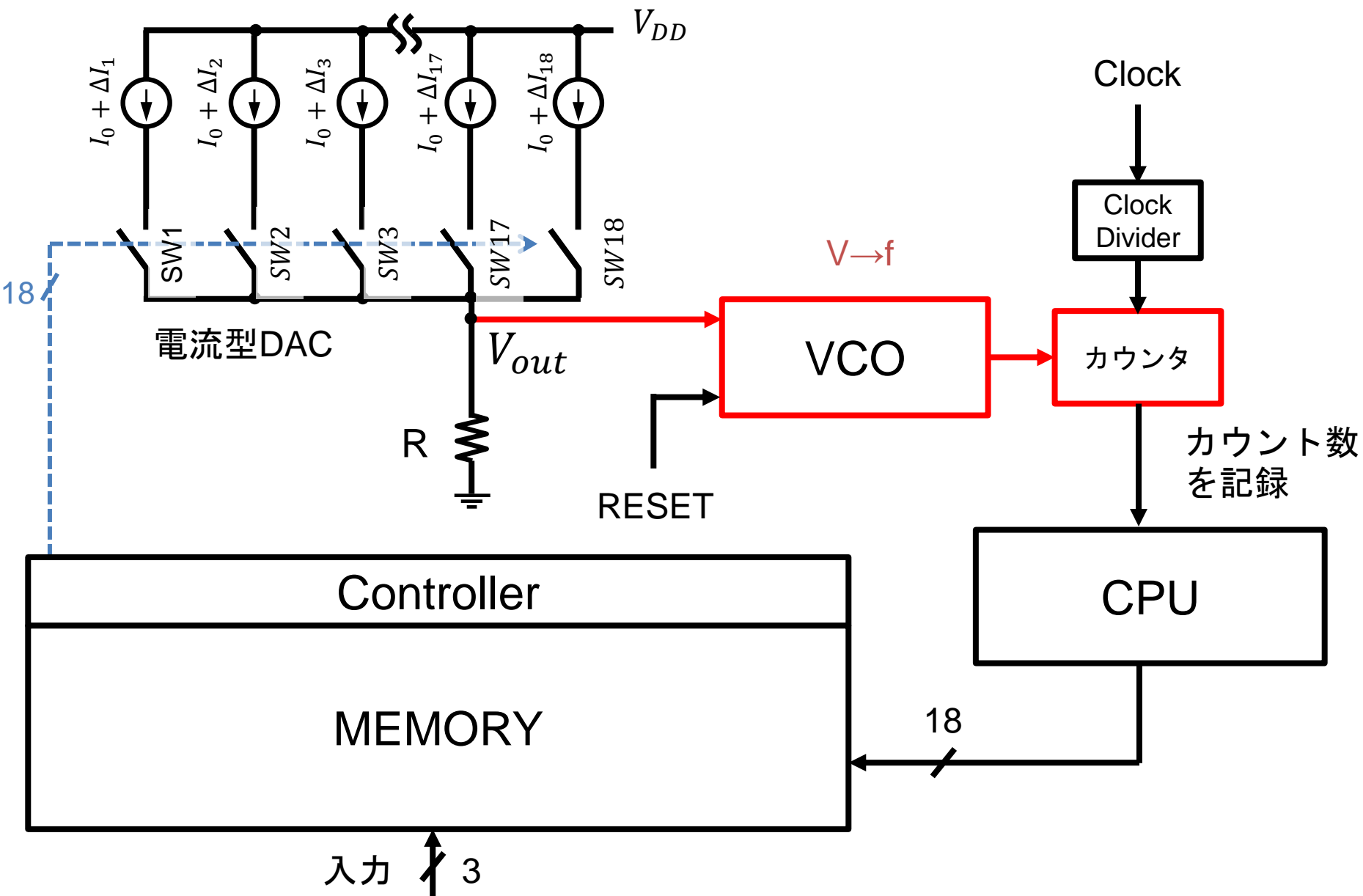


STEP1



STEP2

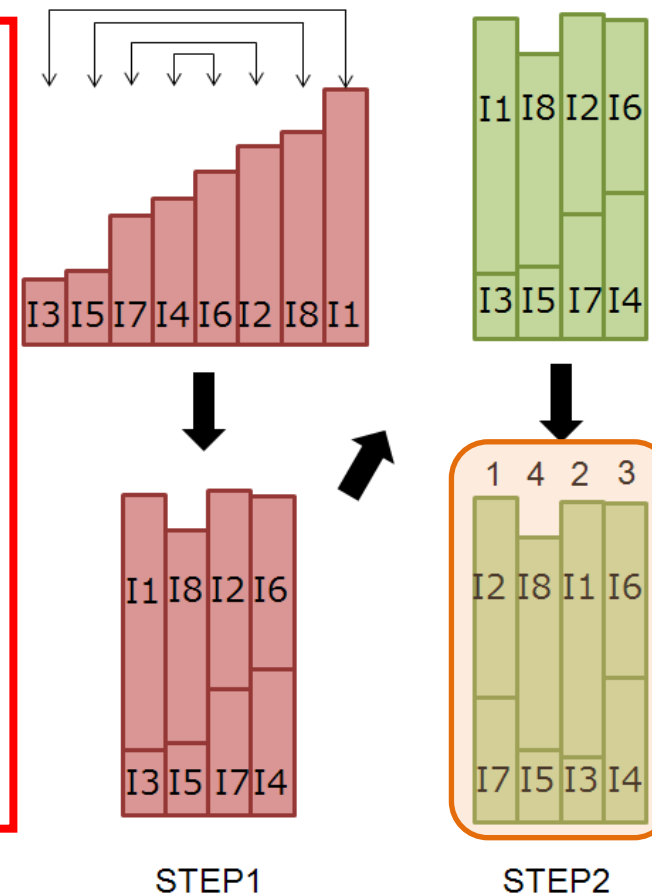
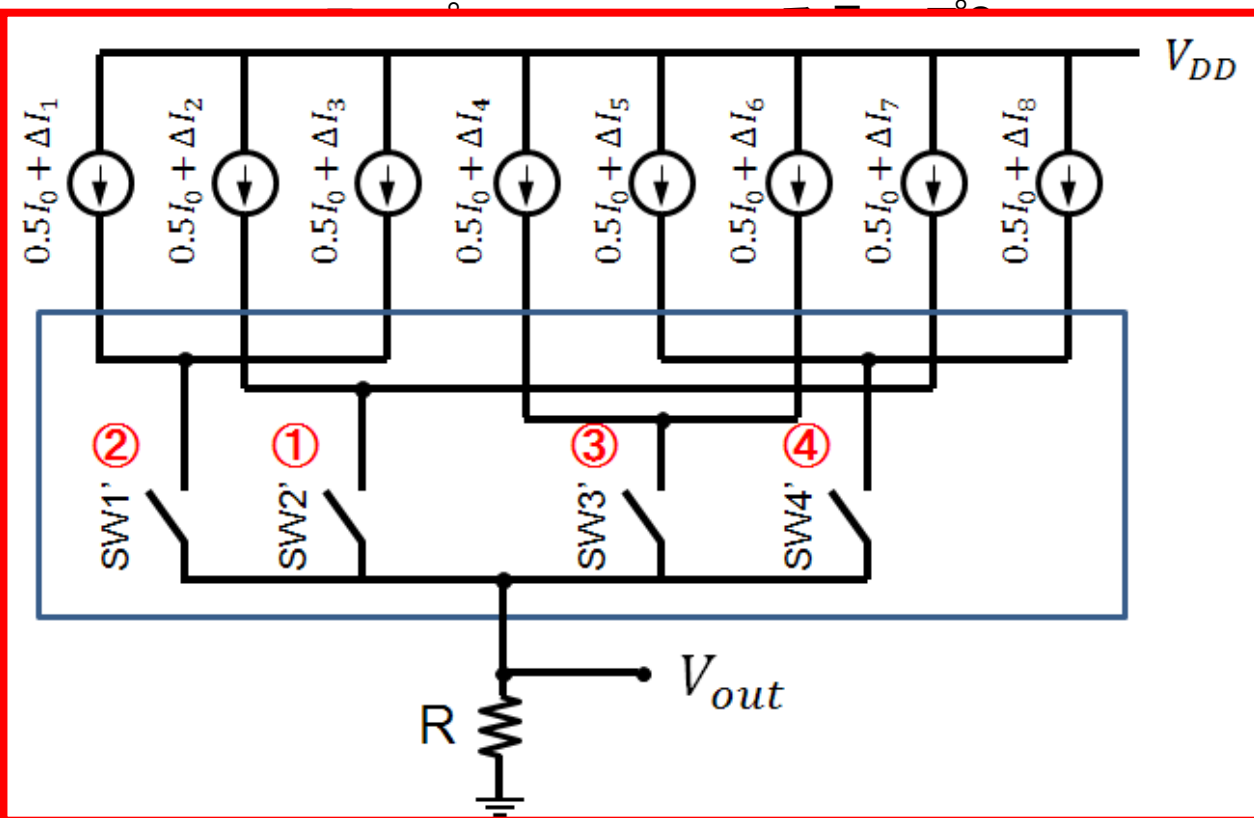
全体図



電流源の並び替え

(6) INLが小になるようONする順番を並び替え

8個を4個にする場合

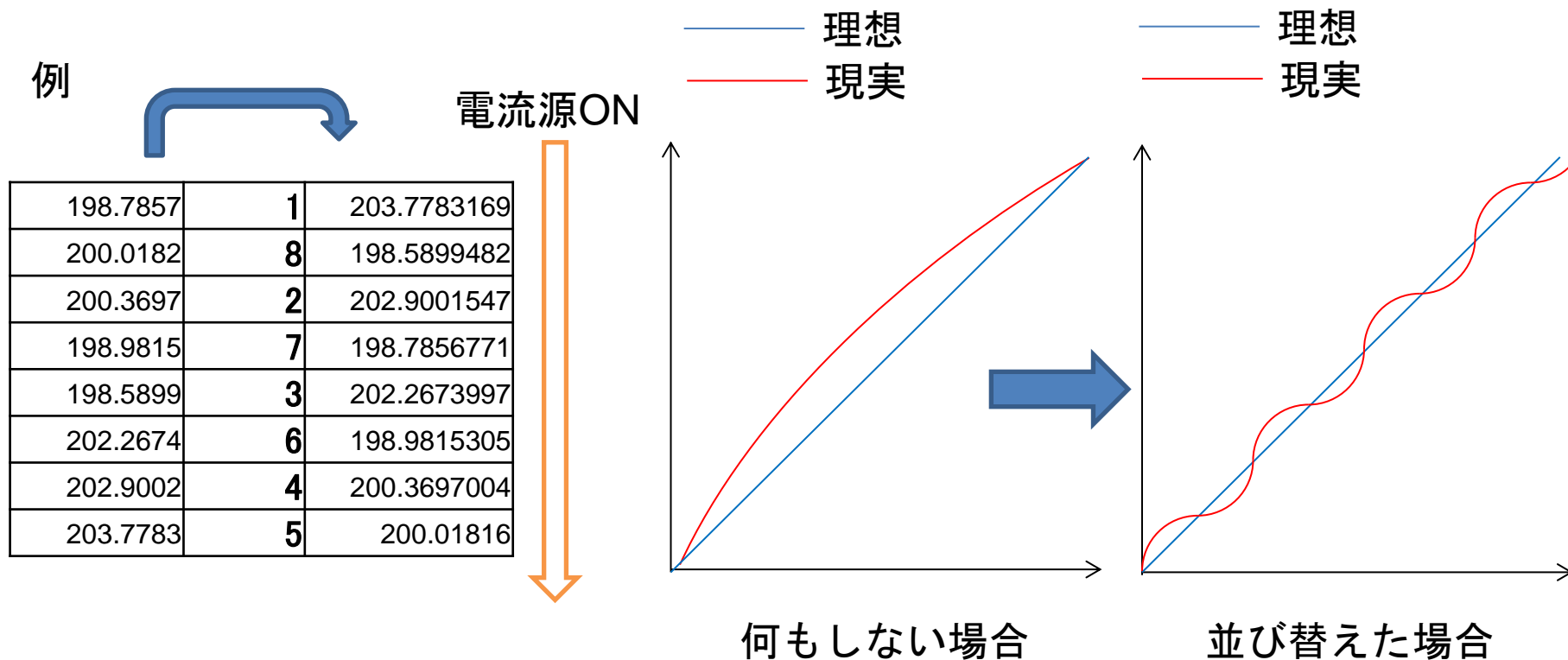


STEP1

STEP2

INL

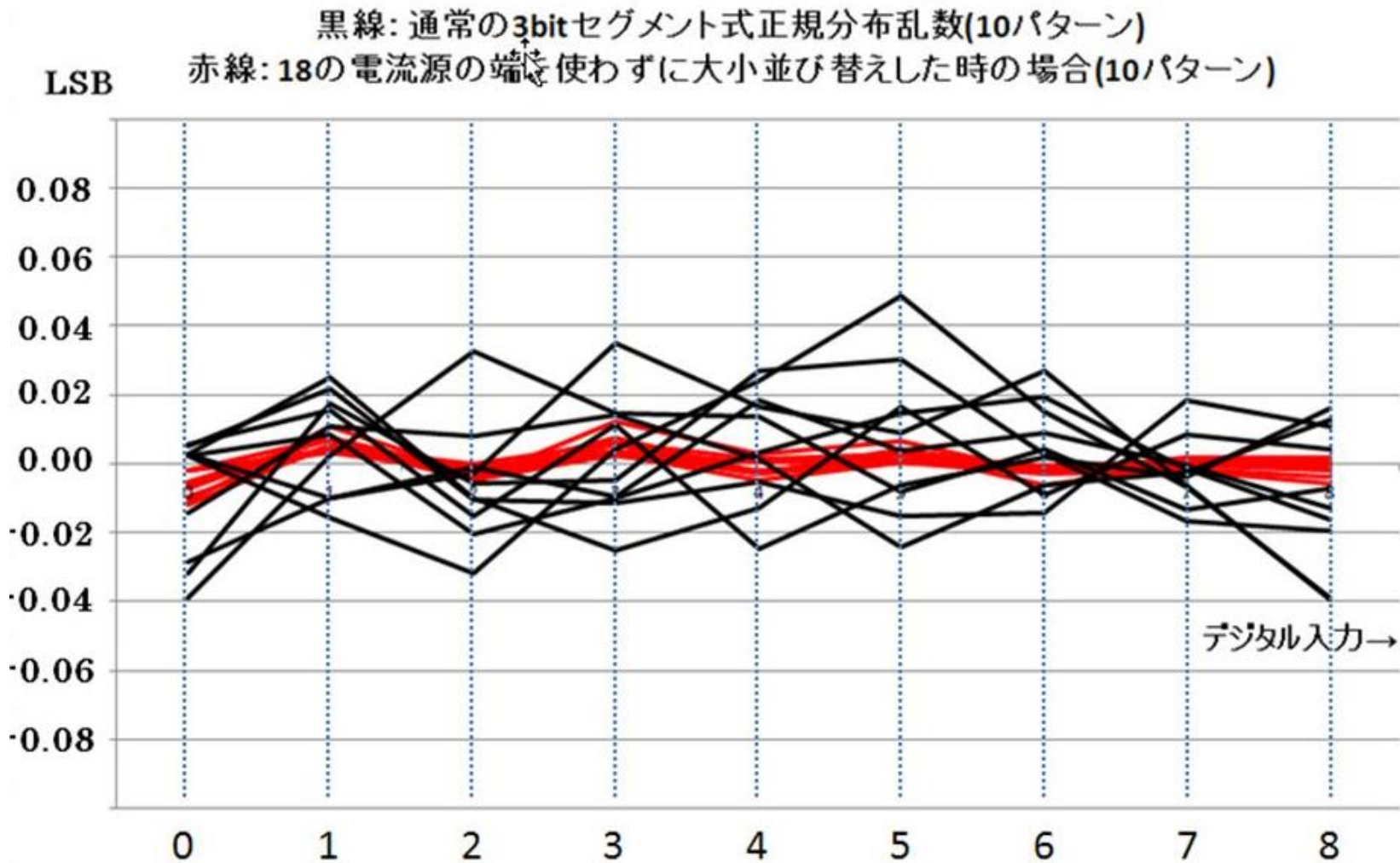
合成後、1番大きい→1番小さい→2番目に大きい→...とし、INLを向上させる



ここでのINLの定義は、ベストフィット積分リニアリティ誤差とする。

INL : 積分非線形性(Integral non linearity)

INL結果



1/3以上のINLの減少効果

電流源の並び替え

(7)並び替え情報メモリに書込み

スイッチ制御

ステップ1

ステップ2

VCOで各素子の電流量を発振周波数に変換

カウンタで発振周波数を測定

内部のCPUを使って各電流源(8素子)を**組合せ**

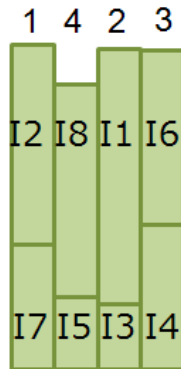
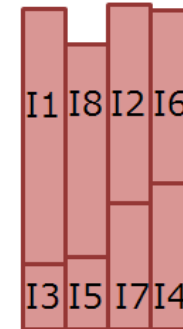
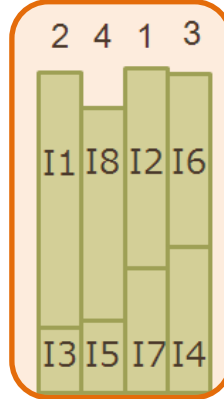
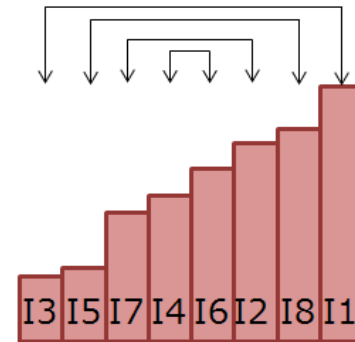
内部のCPUを使って各電流源(4素子)を**並び替え**

メモリに内容を書き込む

ステップ2へ

補正完了

8個を4個にする場合！



STEP1

STEP2

メモリに書き込む(ステップ2)

INLを小さくするよう更に並び替え

並び替え合成後	順位	線形に並び替え
2位+17位 203.4317	1	203.4317
3位+16位 199.6009	8	198.6282
4位+15位 198.6282	2	200.1596
5位+14位 199.1738	7	199.1738
6位+13位 199.7007	3	200.0164
7位+12位 200.0164	6	199.1882
8位+11位 200.1596	4	199.7007
9位+10位 199.1882	5	199.6009

この順に0

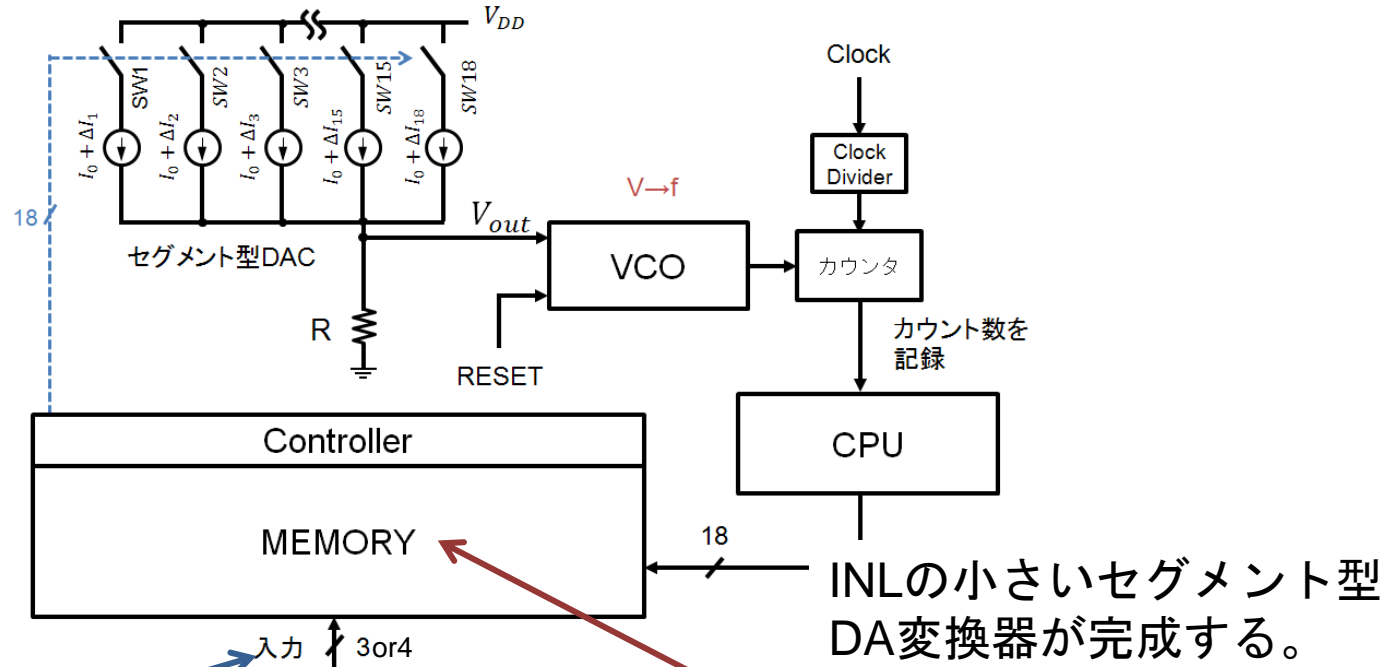
Data IN	A3	A2	A1	A0	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	I12	I13	I14	I15	I16	I17	I18	
0	0	0	0	0																			
1	0	0	0	1					1					1									
2	0	0	1	0							1					1							
3	0	0	1	1								1									1		
4	0	1	0	0				1												1			
5	0	1	0	1	1					1													
6	0	1	1	0															1				1
7	0	1	1	1													1			1			
8	1	0	0	0				1						1									

1位
5位
8位
7位
4位
3位
2位
6位

Data IN	A3	A2	A1	A0	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	I12	I13	I14	I15	I16	I17	I18	
0	0	0	0	0																			
1	0	0	0	1					1					1									
2	0	0	1	0					1			1		1								1	
3	0	0	1	1					1			1		1			1			1	1		
4	0	1	0	0				1	1			1		1			1		1	1	1		
5	0	1	0	1				1	1			1		1			1	1	1	1	1	1	1
6	0	1	1	0				1	1	1		1	1	1			1	1	1	1	1	1	1
7	0	1	1	1	1			1	1	1	1	1	1	1			1	1	1	1	1	1	1
8	1	0	0	0	1			1	1	1	1	1	1	1			1	1	1	1	1	1	1

更に書き換え

補正完了後



Data IN	A3	A2	A1	A0	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	I12	I13	I14	I15	I16	I17	I18	
0	0	0	0	0																			
1	0	0	0	1					1					1									
2	0	0	1	0					1			1		1								1	
3	0	0	1	1					1			1		1			1				1	1	
4	0	1	0	0				1	1			1		1			1		1	1	1	1	
5	0	1	0	1				1	1			1		1			1	1	1	1	1	1	1
6	0	1	1	0			1	1	1			1	1	1			1	1	1	1	1	1	1
7	0	1	1	1		1		1	1	1		1	1	1			1	1	1	1	1	1	1
8	1	0	0	0	1		1	1	1	1	1	1	1	1			1	1	1	1	1	1	1

OUTLINE

- 研究背景
- 研究目的
- アプローチ
- まとめ

まとめ・今後の課題

- 電流源ミスマッチによるDA変換器の非線形性を低減するデジタル手法を提案
- VCOで電流源の大小関係を測定
- 2段階での電流源並び替え
- 数値シミュレーションでの効果を検証

今後の課題

- 実アプリケーションへの対応
(マルチビット $\Sigma\Delta$ ADC内DAC、マルチビット $\Sigma\Delta$ DAC、マルチビット $\Sigma\Delta$ TDCなど)

ご清聴有難うございました。

Q&A

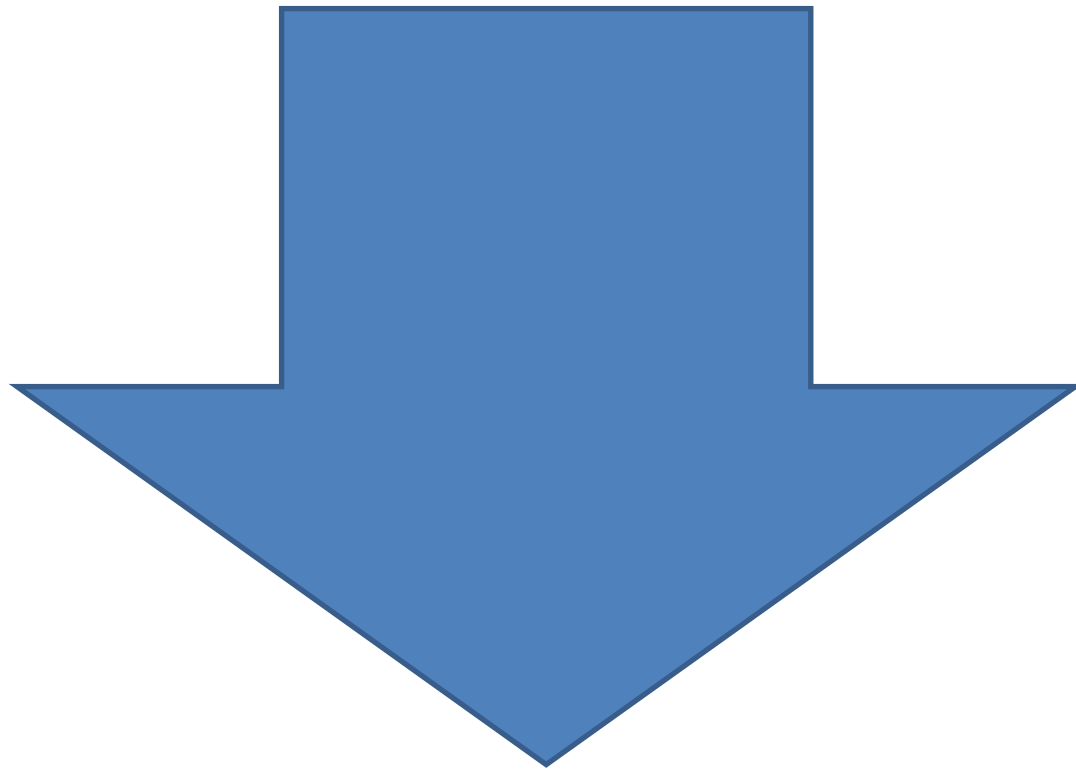
Q.回路面積が増大しますか？

A.増大します。しかし、測定系が小さくでき、並び替えや合成作業は内部CPUを使うと仮定している為、極力小さくできると考えている。

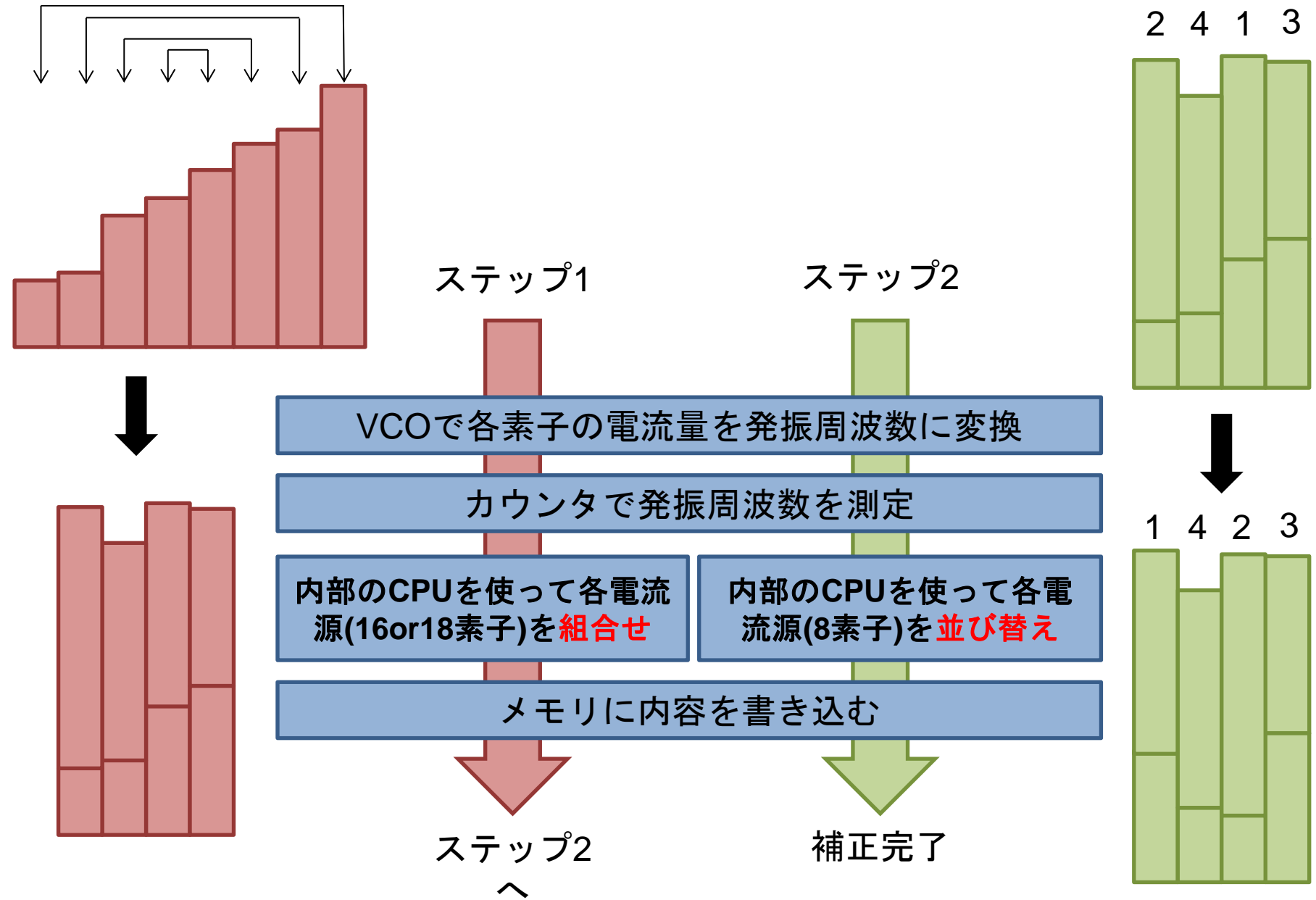
Q.並び替えた時のPVTばらつきの影響はどうか？

A.変動すると思います。これから連続的に校正できるような構成も課題になります。

素材興場



キャリブレーションフロー



全体図

