

基礎電子情報理工学 I

電子工学と情報数理工学の融合(2)

剰余系（孫子算経）による 時間デジタル変換回路アーキテクチャ 設計

小林春夫

群馬大学大学院理工学府 電子情報部門

koba@gunma-u.ac.jp

下記から講義使用 pdfファイルをダウンロードしてください。

出席・講義感想、レポートもここから入力してください。

<https://kobaweb.ei.st.gunma-u.ac.jp/lecture/lecture.html>



レポート提出

この講義の内容に関係したことを調べ
その内容について A4用紙2枚程度にまとめよ。
できるだけ手書きでなくコンピュータを用いよ。

レポートファイル名は 学籍番号(名前)剰余系.pdf にしてください。
たとえば 学籍番号 T201D222 名前 群馬太郎 の場合
T201D222(群馬太郎)剰余系.pdf

提出先(電子提出)、締め切りは下記参照

<https://kobaweb.ei.st.gunma-u.ac.jp/lecture/lecture.html>

台湾訪問

群馬大学大学院 工学研究科 電気電子工学専攻の下記4名が
2012年5月12日(土) - 18日(金)に台湾を訪問した。

安部文隆 (博士前期課程1年)

平林大樹 (博士前期課程1年)

新津葵一 (助教)

小林春夫 (教授、文責)

(I) 台北市 Ambassador Hotel での開催の

IEEE International Mixed-Signals, Sensors, and Systems Test Workshop
(IMS3TW12) にて 大学院生の安部、平林が下記の論文発表を行った。

- [1] Keisuke Kato, Fumitaka Abe, Kazuyuki Wakabayashi, Takafumi Yamada, Haruo Kobayashi, Osamu Kobayashi, Kiichi Niitsu
"Low-IMD Two-Tone Signal Generation for ADC Testing",
- [2] Satoshi Uemori, Masamichi Ishii, Haruo Kobayashi, Yuta Doi,
Osamu Kobayashi, Tatsuji Matsuura, Kiichi Niitsu, Fumitaka Abe,
Daiki Hirabayashi, "Multi-bit Sigma-Delta TDC Architecture for Digital
Signal Timing Measurement"

IMS3TW は参加者40 - 50名程度のアナログ/RF/ミクスドシグナル IC テスト分野のワークショップで、今年で2回目の参加である。(昨年は米国カルフォルニア州サンタバーバラで開催。) 今年も米国、カナダ、欧州(仏、蘭、英、西他) 台湾、日本から参加・発表があった。

アナログ/RF/ミクスドシグナル IC テスト分野の学会・研究者は回路設計の学会・研究者とは異なる。例えば IEEE では回路設計は Circuits and Systems Society, Solid-State Circuits Society になるが、テストは Computer Society である。アナログ/RF/ミクスドシグナル IC テスト分野でまだまだ知らない情報が得られたと同時に、こちらの研究(STARCとの共同研究成果)の、この分野の研究者へのアピールになった。

2012年3月にドイツのドレスデンでの DATE (Design, Automation & Test in Europe) に参加した。LSI テストは設計、検証、EDA、信頼性、診断、歩留まり等とともに総合的に考える必要があると 今回も改めて思った。

IMS3TW12@台北での安部文隆君の発表



IMS3TW12@台北での平林大樹君の発表



(II) 国立台湾大学 (National Taiwan University : NTU) 訪問

IMS3TW12 のプログラム委員長 Prof. Jiun-Lang Huang (黄俊郎先生)の研究室を訪問させていただいた。NTU は台湾での最高峰の大学との社会的評価がある。

その中で電気電子工学分野は非常に高い人気とのことである。台湾は国策としてエレクトロニクス、半導体に「選択と集中」している。かつてはNTUではIC設計分野はほとんどなかったが、現在はこの分野で世界のトップの大学の一つである。**From Zero to Hero (ゼロからヒーローへ)** の紹介が印象に残った。

同大学の Graduate Institute of Electronics Engineering (電子工学研究所)では教員 47 名、修士課程学生 447 名、博士課程学生 212 名である。学生の就職先は TSMC, MediaTek 等の台湾の半導体関連メーカーが多いとのことである。

NTU に対して、国や産業界は特別な扱いをしているような印象を受けた。MediaTek, Intel, IBM は NTU に研究所を設立している。黄俊郎先生はいくつもの会社や国立の研究所 (Industrial Technology Research Institute of Taiwan: ITRI) と共同研究を行ってきている。

米国等の海外の大学院への進学は、かつては 8 割程度であったが、現在は 2 割程度であり、米国の大学での電気電子工学分野の台湾出身の教授が相対的に減少している。



黄俊郎先生 (左から 3 番目) に招待され、NTU 大学院生を交えての昼食会

(III) 国立交通大学 National Chiao Tung University 訪問

IMS3TW で知り合った国立交通大学 Prof. Hao-Chiao Hong (洪浩喬先生、ADC/DAC 設計とその BIST の研究、TSMC 社出身) の研究室を訪問させていただいた。同大学のある新竹 (HsinChu) 市は 台北から高速列車(新幹線)を利用して30分で、近代的な街であり風が強い。このリサーチパークに同大学がある。電気電子工学科は約80名の教員がおり (そのうち日本人2名)、台湾最大の電気電子工学科とのことである。

洪浩喬先生のご案内でこのリサーチパークの国立 Chip Implementation Center (CIC)も訪問し、Ms. Shuw-Guann Lin (林劭冠 先生)に高周波回路測定関係の設備紹介を受けた。

このリサーチパークには 国立精華大、TSMC, MediaTek, RealTek, ITRI 等の半導体関係の大学、企業、研究所が集積している。



洪浩喬先生 (右から5番目) の研究室にて。右から6番目は Prof. Yi Chiu (邱一 先生, MEMS 関係の研究)

(IV) 最後に

台湾社会の悩みは少子化とのことである。台湾はパッケージング、組立で優位を保っているが IC 設計では中国の追い上げにあっている。

いずれにせよ、日本の国益を考えるなら「日本の。。。」という発想を超えなければならぬという思いを強くした。

台北の夜市 (Night Market)



台北は 物価が(日本に比べて)安い、日本人観光客多い



国立伝統芸術中心 (National Center for Traditional Arts)



台北 101, 台湾科学技術大学, 国父記念館 (孫文記念館)、博物館等も訪ずれた。

台北(台湾)出張報告

2012.5/12~5/18

群馬大学工学研究科 電気電子工学専攻
安部文隆 平林大樹

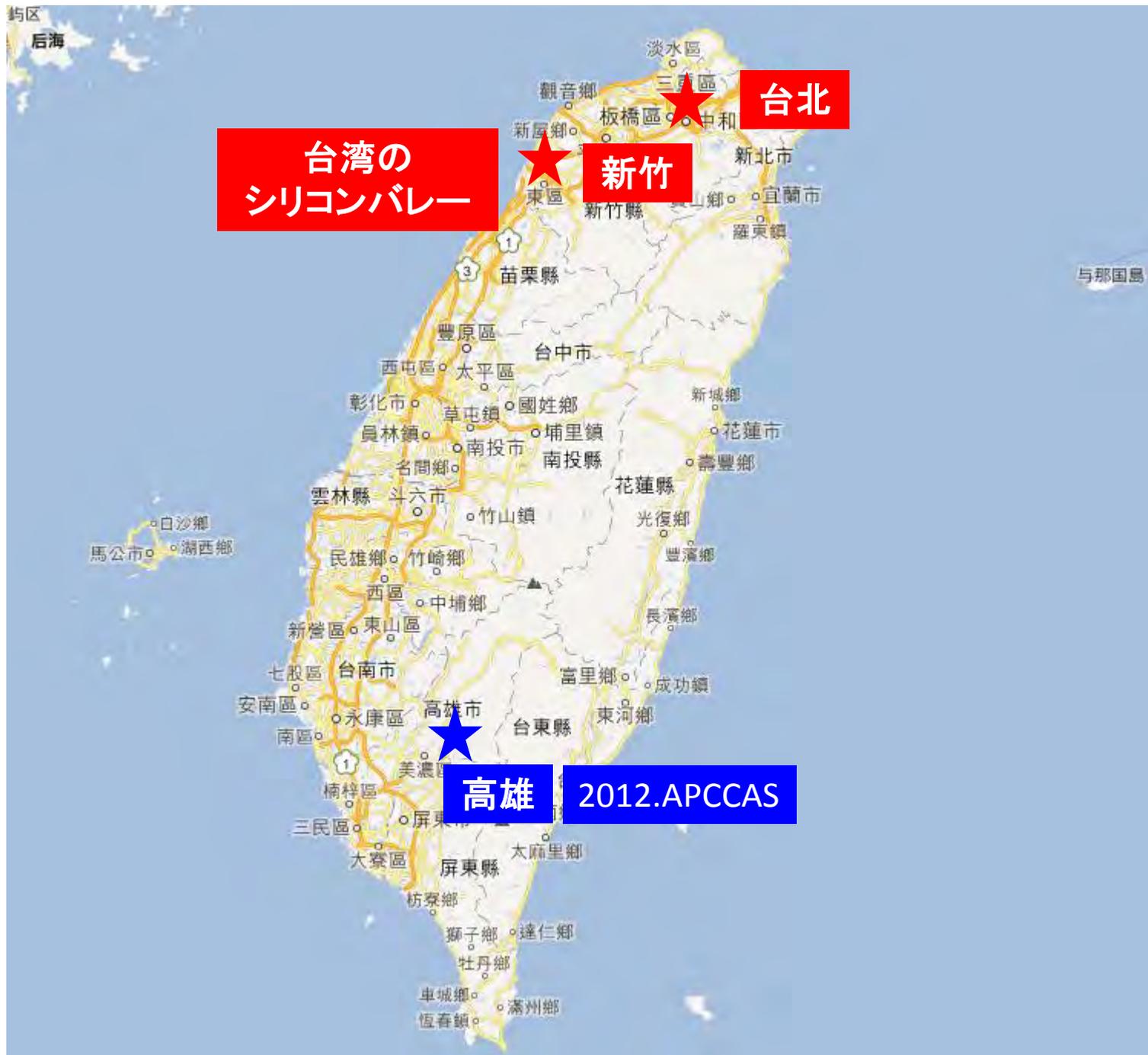
台灣旅行記

渡航期間

2012.5/12~5/18

渡航目的

- IMS3TW'12 参加・発表
- National Taiwan University訪問
- National Chiao Tung University訪問



台湾の
シリコンバレー

新竹

台北

高雄 2012.APCCAS

羽田国際線ターミナル



きれいな空



台湾の夜



国立故宮博物館へ



国立故宮博物館



写真を撮る平林と新津先生



白菜と肉形石を食べに





牛肉麵



肉形石(豚の角煮)これはおいしい^^



白菜これもおいしい^^



101にいったらいいじゃん。



101へ上ります。NT \$ 400(学割)。



眺めよいですよ。



黄色：タクシー。たくさん走ってます。



珊瑚でできた置物に喜びを隠しきれない安部



平林 & 安部



平林



解説をしてくれます。



101の屋上で



101の屋上で



鷲？鷹？

どー思う？平林。

そっかー。





荒川君のレノボで遊ぶ新津先生

地上400mのWiFiスポット



真剣に探索する平林

地上400mのWiFiスポット



そしてもう夕方



101の前で



スイーツ食べに行くよ。南国だもん♡



タクシー探し



タクシー乗りたい

タクシー来ねーな。

疲れましたね。



タクシーゲット。中国語話す新津先生



そして、マンゴー+かき氷ゲット



照れる



食べる



そっかー



そしてまたすぐに夕飯

まだ、食べられる？

イケイケです。



お茶



乾杯



メニュー検討平林君新津先生。



カニ



カニを撮る新津先生



カニ



カニ



新津せんせい食べる。



安部



平林



美味しそうに食べる新津先生



安部です。



学会1日目



平林



何を撮っているんだ。



MRT乗ります。I♡台北。



コインの切符



夜市



ナイトマーケット。こんな感じです。



カエル？



☆南国ですから☆



☆フルーツ☆



食べますか？食べます。



自分で自分を撮る。安部。



試食。おいしいです。



えび釣り



シアター発見。



Bubble Tea知っていますか？



タピオカでした。



飲む。



学会2日目。発表を迎えた平林。



とりあえず。コーヒーを。



緊張なう。

行ってきます。



平林君の発表です。①



平林君の発表です。②



平林君の発表です。③



2日目学会無事終了。昼食へ。



いやー、よかったよ。

そうですねー^^

日本料理です。



ちなみに今日は晴れ。台湾暑い。



寿司。すし。SUSHI。



すし、お刺身舞台。



ガルシア。発表終わりました。



魚をとるガルシア平林



安部とガルシア平林



安部と寿司



ケーキをゲットした平林



台湾ビール。



薄味？おいしいです！

明日の発表どうしましょう。





とりあえず食べたらいいよ。

そっかー。

ケーキを食べる。



新竹に行くことになるの巻。

国立交通大学訪問させて下さい。

OK!!



Traditional Arts Center





台湾文化に触れる新津先生。





そーっと。



あめ？



でかい筆。



平林 & 新津先生



夕食なう。



お茶。明日の発表無理ぽ・・・



国立交通大学の先生



学会3日目。



発表安部①



発表安部②



発表終わり喜ぶ安部



国際交流をする小林先生

日本から来ました。宜しくお願いします。



あなたの研究に興味があります！

そっかー！



Kobawebにメールを送る小林先生。

早速Kobawebにメールするよ。クイックが重要だよ！

さすがっす！



学会三日目午前終了後の平林。



終わりました。



台湾4日目にしてようやく小籠包のお店へ



メニューに迷う新津先生とガルシア



次の目的地に迷う新津先生



たれの作り方を新津先生に教わる。



二刀流。



こんなに頼んだ。我々。



写真を撮る新津先生。目に焼き付けるガルシア平林。



安部と小龍包。



台北のマップルで次の観光地は



この後更にデザートの小籠包頼みました。



新津先生のwifiと荒川君のレノボのコラボレーション



アクセス成功！



おやつ探しの巻



試食しましたか？

このあめ研究室の皆に配ります。

名案だね!!!!



会計をする平林君

これで研究室のみんなも喜ぶ!!



NTU訪問



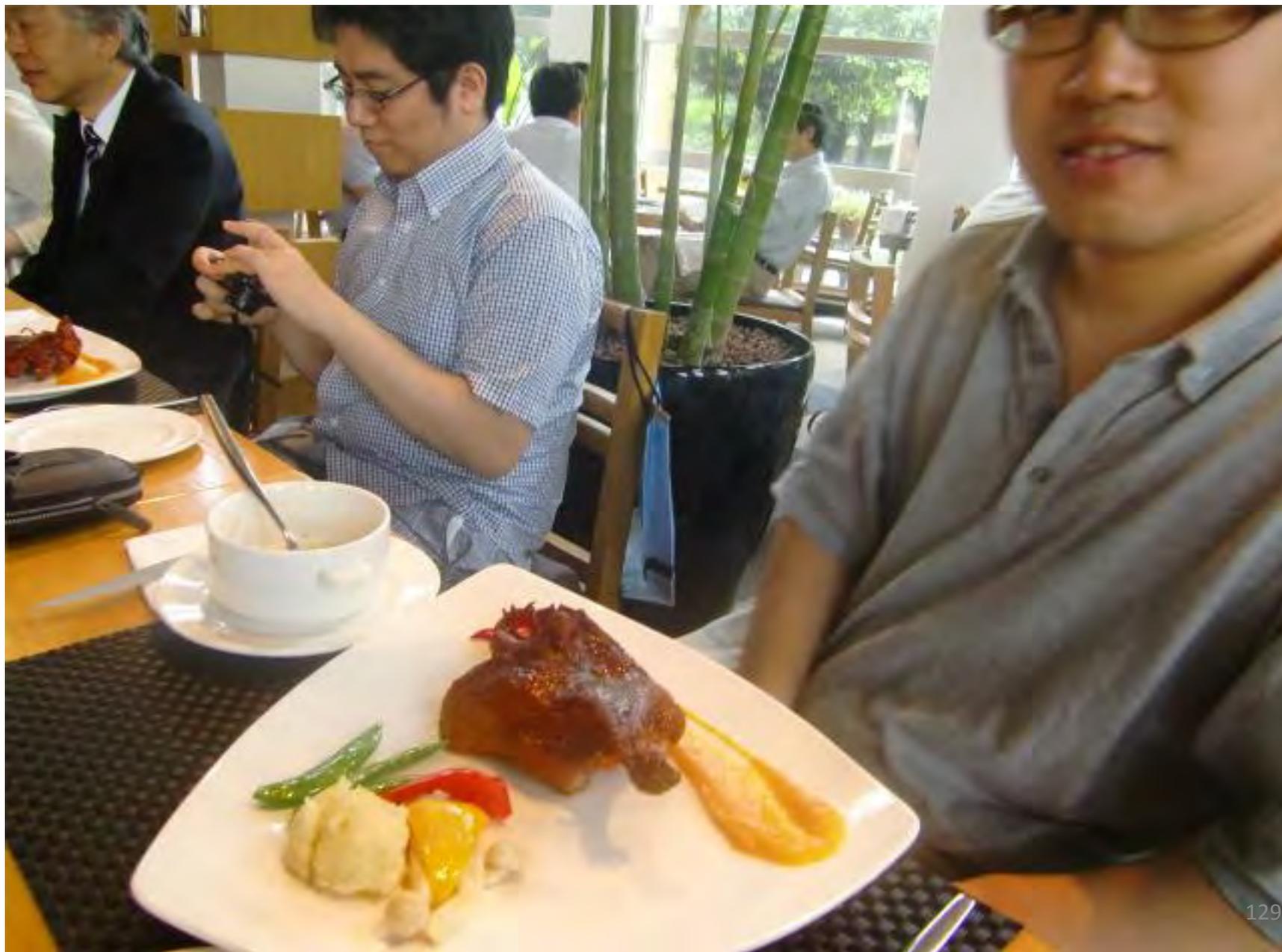
群馬大学紹介@NTU



みんなで食事@NTU



肉食系男子



NTUの学生、先生と記念に一枚



From 台北 to 新竹



新津先生、小林先生



群馬大学紹介@国立交通大学

エネルギーハーベスト！

そっかー！



国立交通大学



国立交通大学の学生

日本の
アニメ大好き



新竹を去る平林ガルシア新幹線乗ります



新幹線_台湾バージョン



最後の珊瑚_安部



最後の珊瑚_平林



さらば台湾





剰余系(孫子算経)を用いた 時間デジタル変換回路

群馬大学大学院 理工学府 電子情報部門
小林春夫

koba@gunma-u.ac.jp

<https://kobaweb.ei.st.gunma-u.ac.jp/>

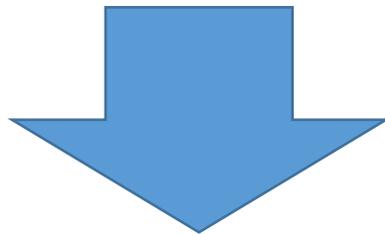
講義資料: <https://kobaweb.ei.st.gunma-u.ac.jp/lecture/lecture.html>

中国の剰余定理

- 中国の算術書『孫子算経』

「3で割ると2余り、5で割ると3余り、7で割ると2余る数はいくらか」

一般化



中国の剰余定理

答え 23

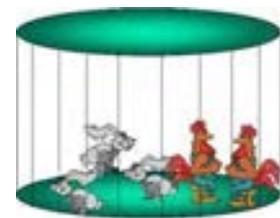


孫子算経

孫子算經

- 「3で割ると2余り、5で割ると3余り、7で割ると2余る数は何か」 答え 23
→ 一般化したのが「**中国人の剰余定理**」。

- **鶏兔同籠(けいとどうりゅう)**
「キジとウサギが同じ籠(かご)。頭が35個
足は94本。キジ、ウサギはそれぞれいくらか。」



→ 日本に入ってきて「**鶴亀算**」となる

- が、孫子算經と孫子兵法とは
直接は関係ないようである。

二人の孫子

「孫武」

戦わずして勝つ



「孫臏 (そんびん)」

馬を三組ずつ出して勝負する競馬。

相手の上等の馬が出る競走に自分の下等の馬、

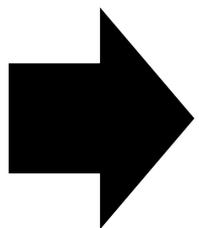
中等の馬が出る競走に上等の馬、

下等の馬が出る競走に中等の馬を出させる。



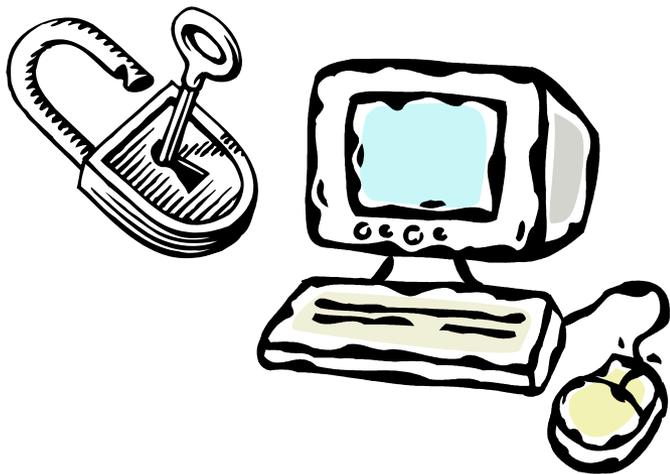
中国の剰余定理のアナログ回路への応用

- ✓ 江戸時代、「百五減算」として伝来
- ✓ 現在、情報セキュリティの暗号化に応用



古典数学によるイノベーション

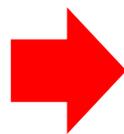
集積回路に応用



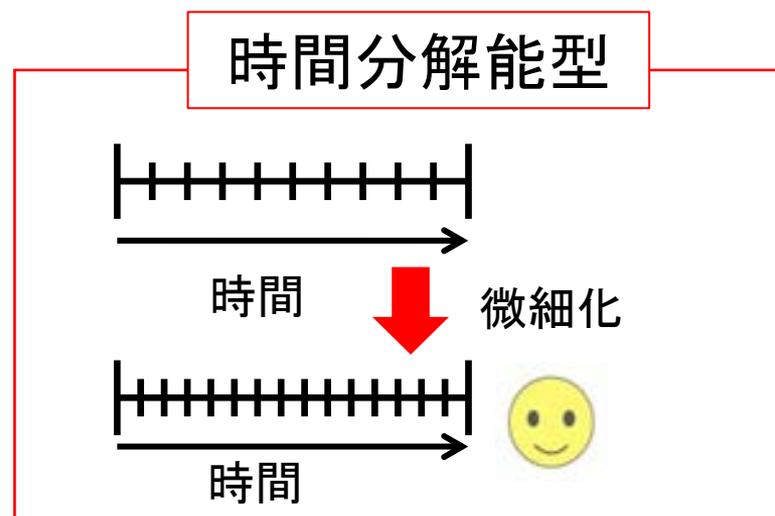
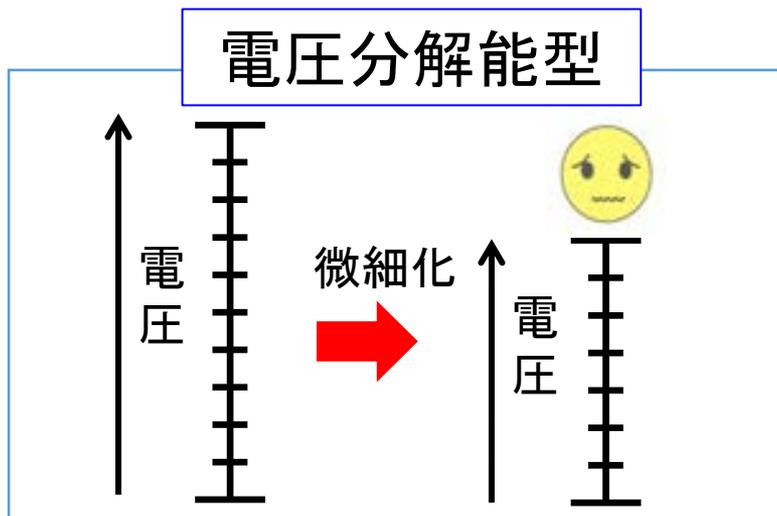
関孝和

研究背景

微細化CMOS LSI



電源電圧の低下
動作スイッチングスピードの向上

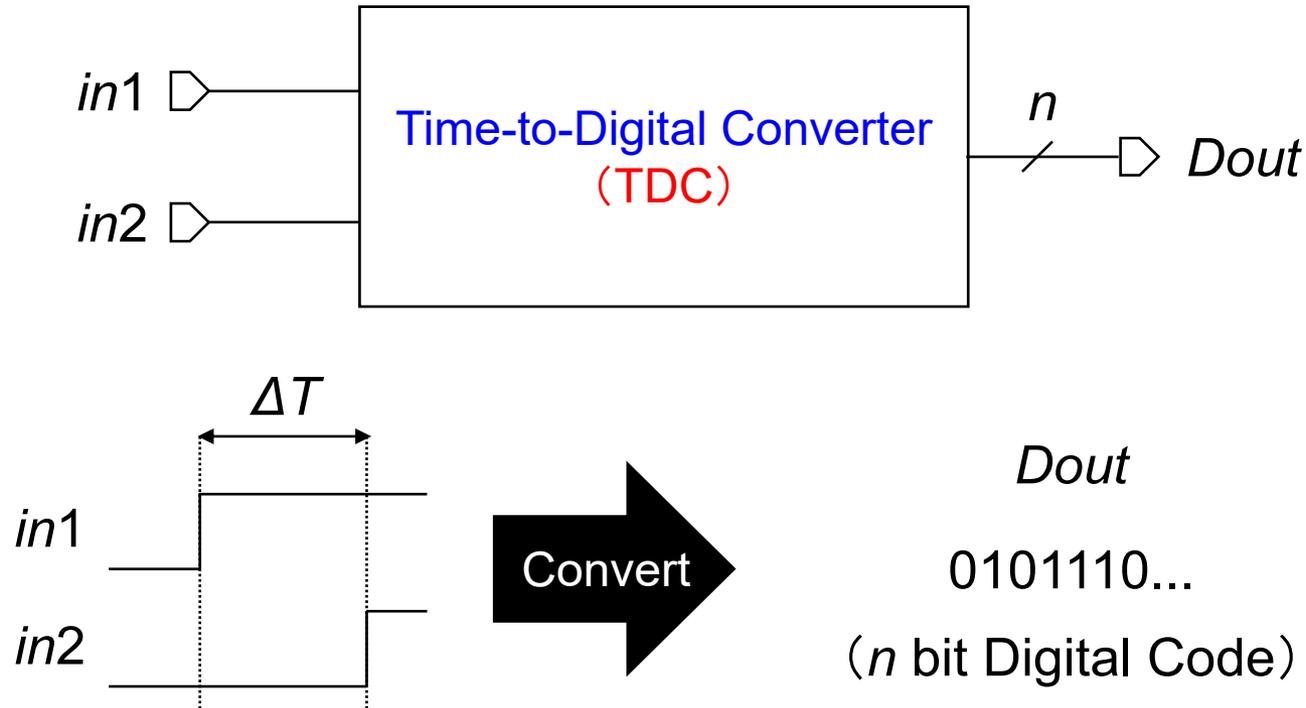


TDC (Time-to-Digital Converter) は2つのデジタル信号の時間差をデジタル値に変換



微細化CMOS LSIにおいて、TDCは時間領域アナログ回路のカギとなる
(センサ回路, All-Digital PLL, ADC, 変調回路等)

タイムデジタイザ回路

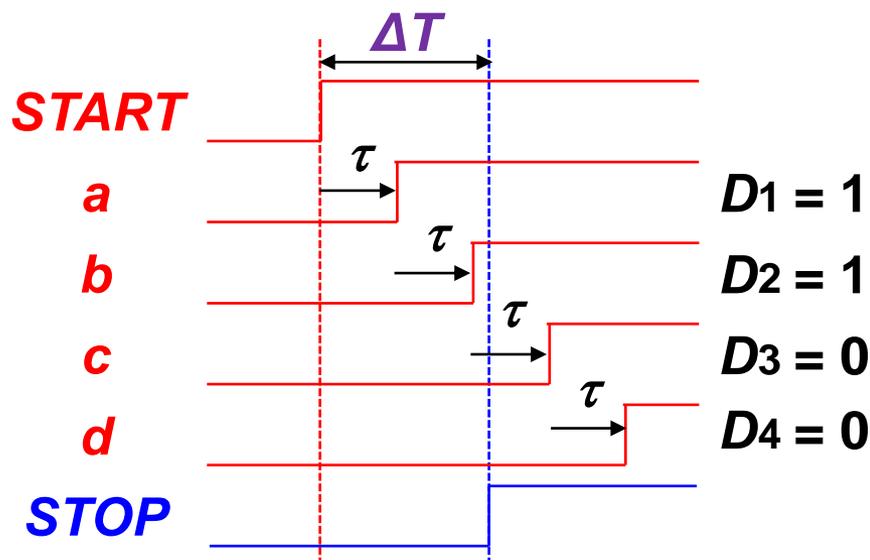
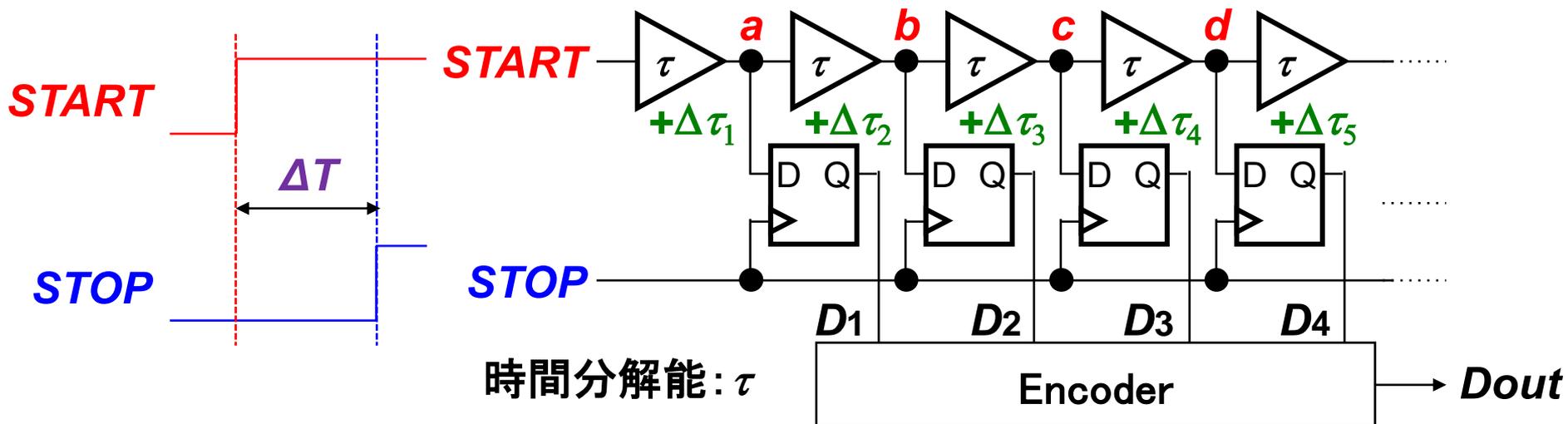


2つのデジタル信号間の時間差 ΔT をデジタル値に変換



出力のデジタル値より ΔT を測定可能

フラッシュ型 TDCの構成と動作



- ΔT の大きさに比例したデジタル値 Dout を出力

- 時間分解能 τ

高エネルギー加速器研究機構
素粒子原子核研究所
新井康夫氏による発明

フラッシュ型TDCの回路規模の問題

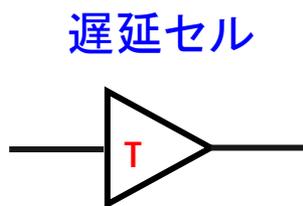
START とSTOP の立ち上がりエッジ間の時間差

測定範囲 $0 < \Delta T < N T$ ΔT

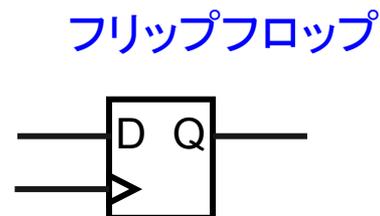
時間分解能 T

$N = 1001$ (千一) のとき

フラッシュ型TDC では大きな回路規模、大きな消費電力



1001個



1001個

提案する剰余系TDC $1001 = 7 \times 11 \times 13$
 同じ測定範囲、時間分解能で $7 + 11 + 13 = 31$ 個の
 遅延セル、フリップフロップで実現できる

千一個から三十一個へ !!

研究の目的

時間測定回路TDC

- LSIテストシステムのキーコンポーネント
- 時間信号であることを利用

 “剰余”が容易に得られる

- 剰余系を利用

フラッシュ型TDCに比べ、
同等性能、小回路規模・低消費電力TDCが
実現できる可能性あり



剰余系TDC回路を検討

剰余系の例

基数 2, 3, 5 互いに素

$$N=2 \times 3 \times 5 = 30$$

0から $N-1(=29)$ までの整数の一つを k

$$a: k \text{ を } 2 \text{ で割った余り} \quad a = \text{mod}_2(k)$$

$$b: k \text{ を } 3 \text{ で割った余り} \quad b = \text{mod}_3(k)$$

$$c: k \text{ を } 5 \text{ で割った余り} \quad c = \text{mod}_5(k)$$

k と (a, b, c) の組は1対1に対応する。

k を (a, b, c) で表現 \longrightarrow 剰余表現

中国人の剰余定理 (Chinese Remainder Theorem)

(a, b, c) から k を求めるアルゴリズム



剰余定理の例

基数 2, 3, 5 互いに素

$$N=2 \times 3 \times 5 = 30$$

0から $N-1(=29)$ までの整数の一つを k

a : k を2で割った余り $a = \text{mod}2(k)$

b : k を3で割った余り $b = \text{mod}3(k)$

c : k を5で割った余り $c = \text{mod}5(k)$

k と (a, b, c) の組は1対1に対応する。

k を (a, b, c) で表現 \rightarrow 剰余表現

剰余定理 (Chinese Remainder Theorem)

(a, b, c) から k を求めるアルゴリズム

剰余定理は、

この問題を他の整数についても適用できるように一般化したもの。

自然数 k と剰余表現 (m_1, m_2, m_3) は 1対1 対応

m_1	m_2	m_3	k
0	0	0	0
1	1	1	1
0	2	2	2
1	0	3	3
0	1	4	4
1	2	0	5
0	0	1	6
1	1	2	7
0	2	3	8
1	0	4	9
0	1	0	10
1	2	1	11
0	0	2	12
1	1	3	13
0	2	4	14

m_1	m_2	m_3	k
1	0	0	15
0	1	1	16
1	2	2	17
0	0	3	18
1	1	4	19
0	2	0	20
1	0	1	21
0	1	2	22
1	2	3	23
0	0	4	24
1	1	0	25
0	2	1	26
1	0	2	27
0	1	3	28
1	2	4	29

剰余DCの原理

TDC 回路は信号が時間であることを利用すると“剰余”が容易に得られる。

三つのリング発振回路(遅延 m_1T , m_2T , m_3T)を利用し、発振状態から経過時間 T の測定を行うことが可能で。剰余定理に基づいて、(a, b, c)から k を求め、経過時間 $T = k \times T$ を得る。

例えば、三つのリング発振回路(遅延 $2T$, $3T$, $5T$)を利用し、発振している状態から経過時間 T の測定を行う。

T を $2T$ で割った余りは a

T を $3T$ で割った余りは b

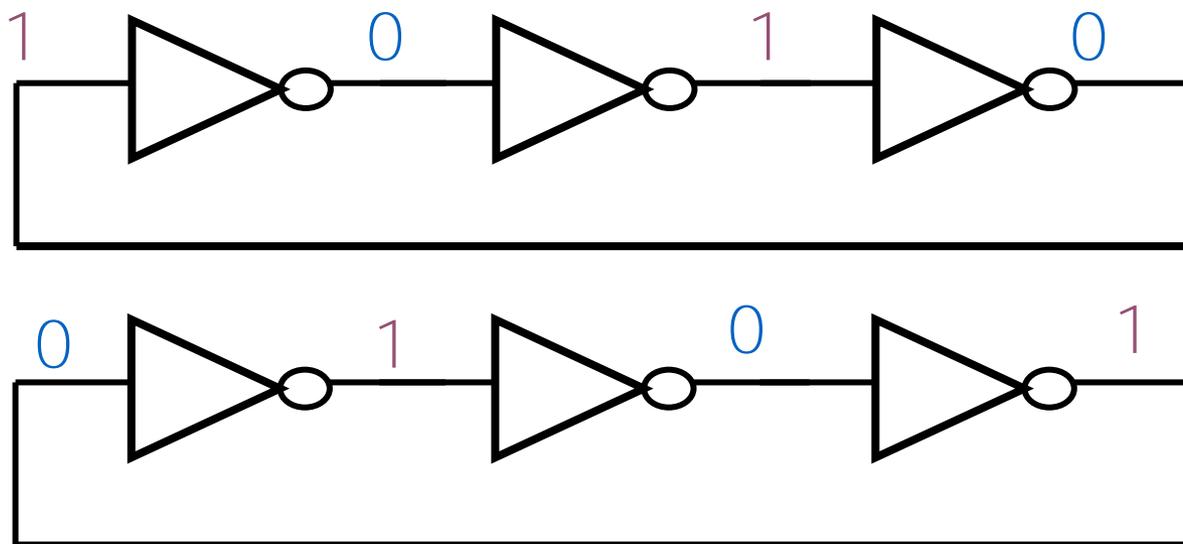
T を $5T$ で割った余りは c

⇒剰余定理で $T = k^*T$

リング発振器 (Ring Oscillator)

奇数個インバータのリング接続

安定状態
なし



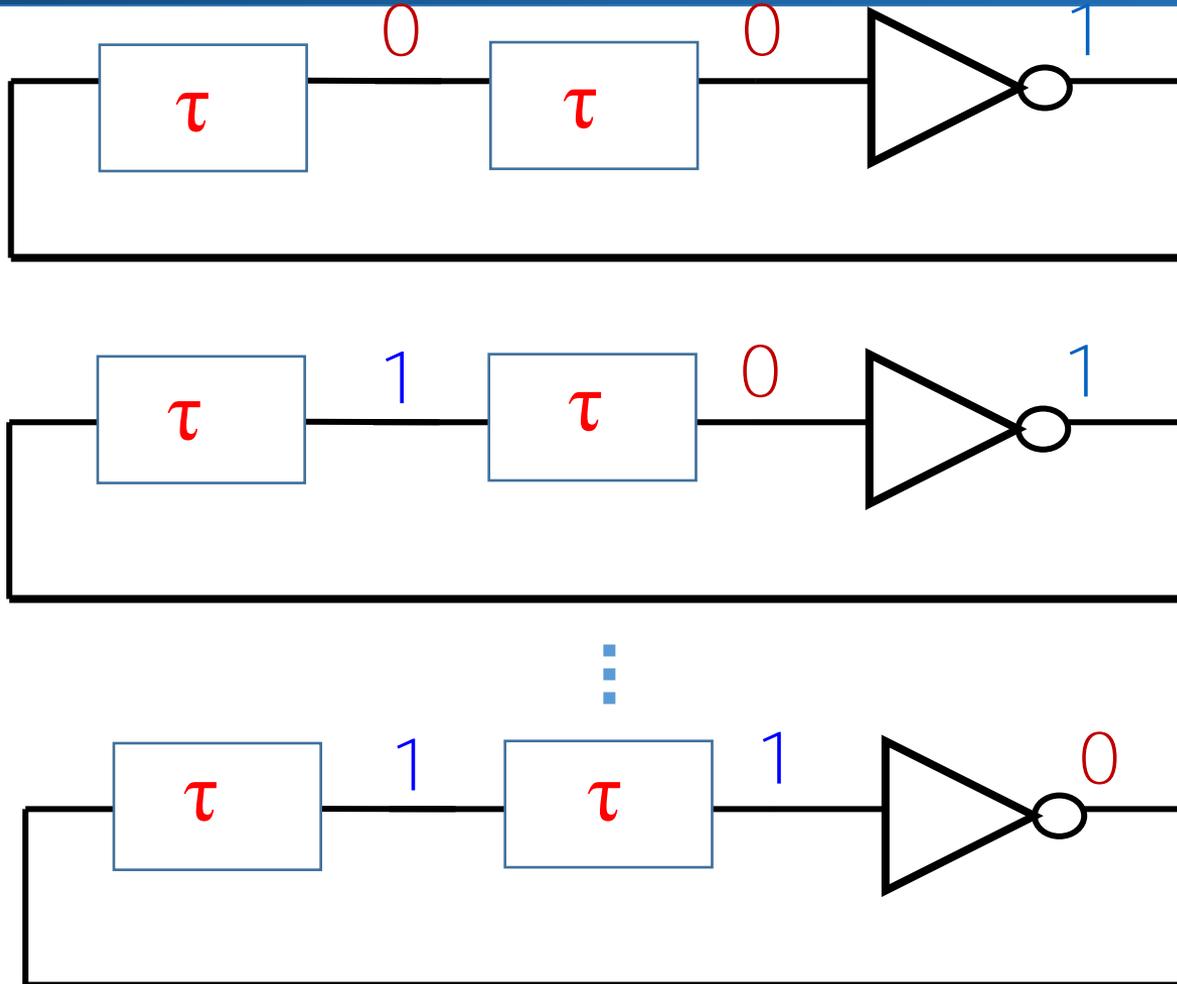
T: インバータ遅延、 $2N+1$ 個のインバータリング接続

周波数 $f = \frac{1}{2(2N+1)T}$ で発振。



メビウスの帯

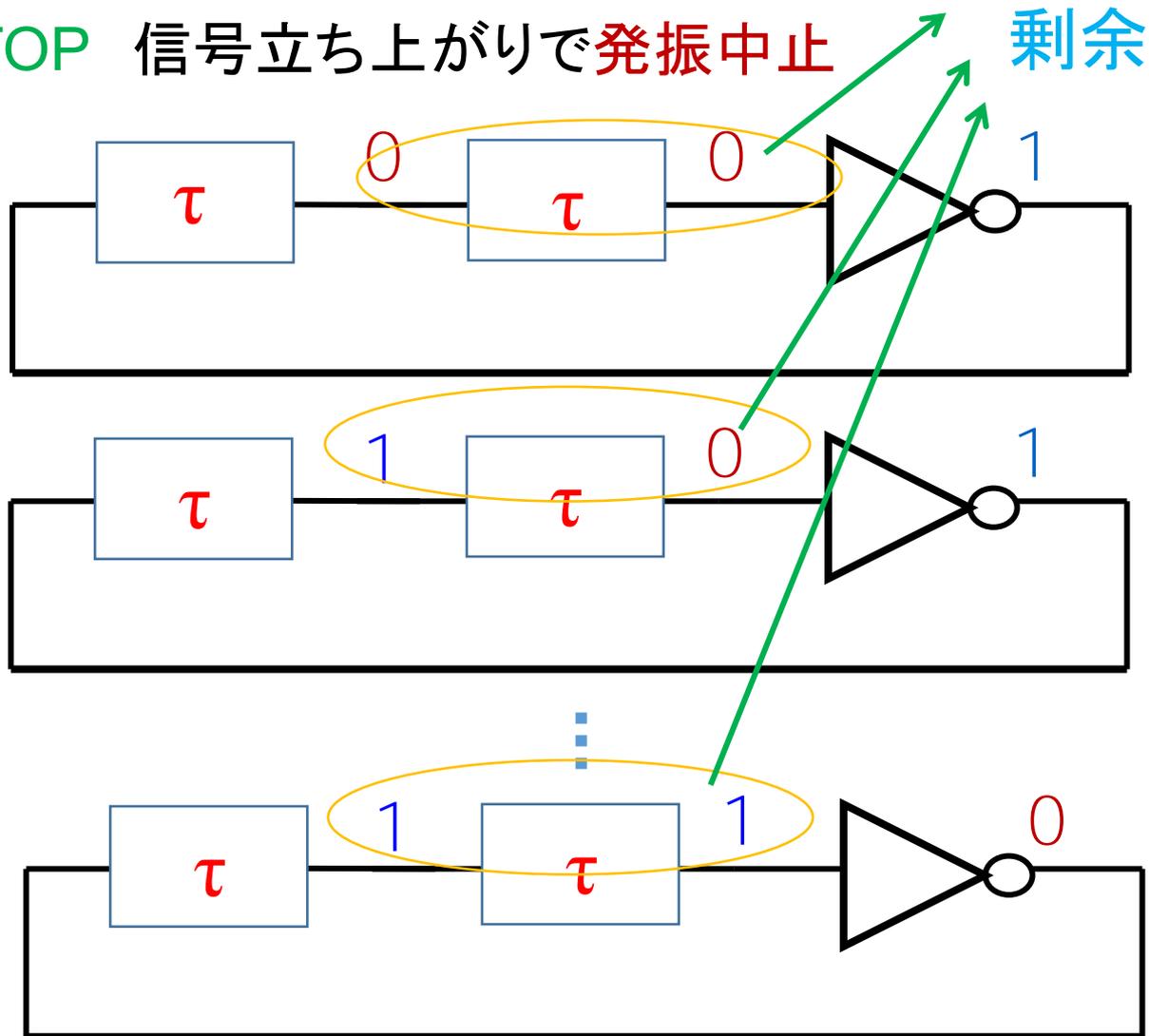
リング発振回路で剰余が容易に得られる



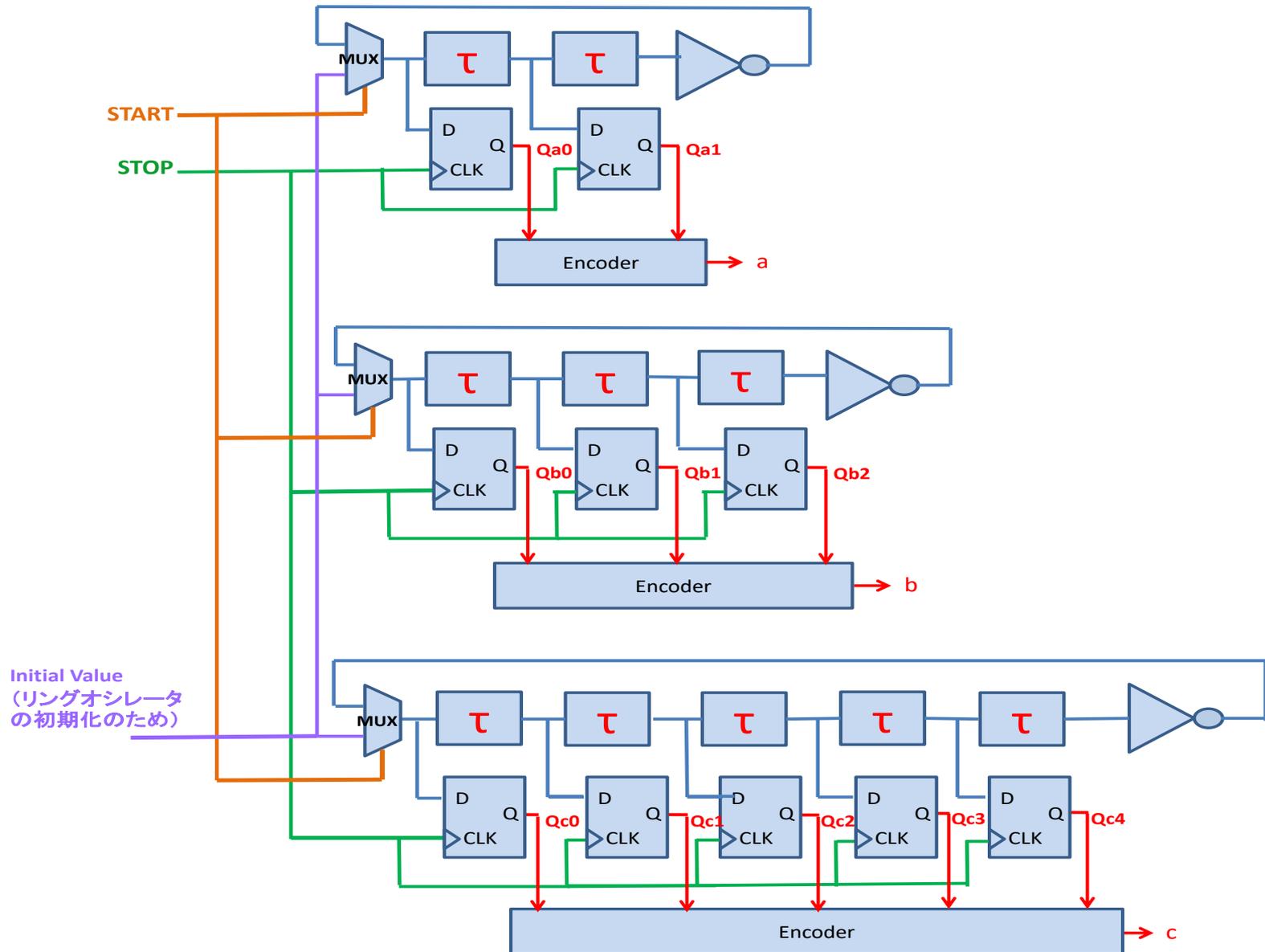
考察 TDCでは取り扱う入力信号が時間信号なのでリング発振回路構成により剰余が容易に得られる。電圧信号を入力とするADCでは剰余を得るのは簡単ではない。

リング発振回路で剰余を得る

- **START** 信号立ち上がりで発振開始
- **STOP** 信号立ち上がりで発振中止



提案する剰余系TDCの回路図



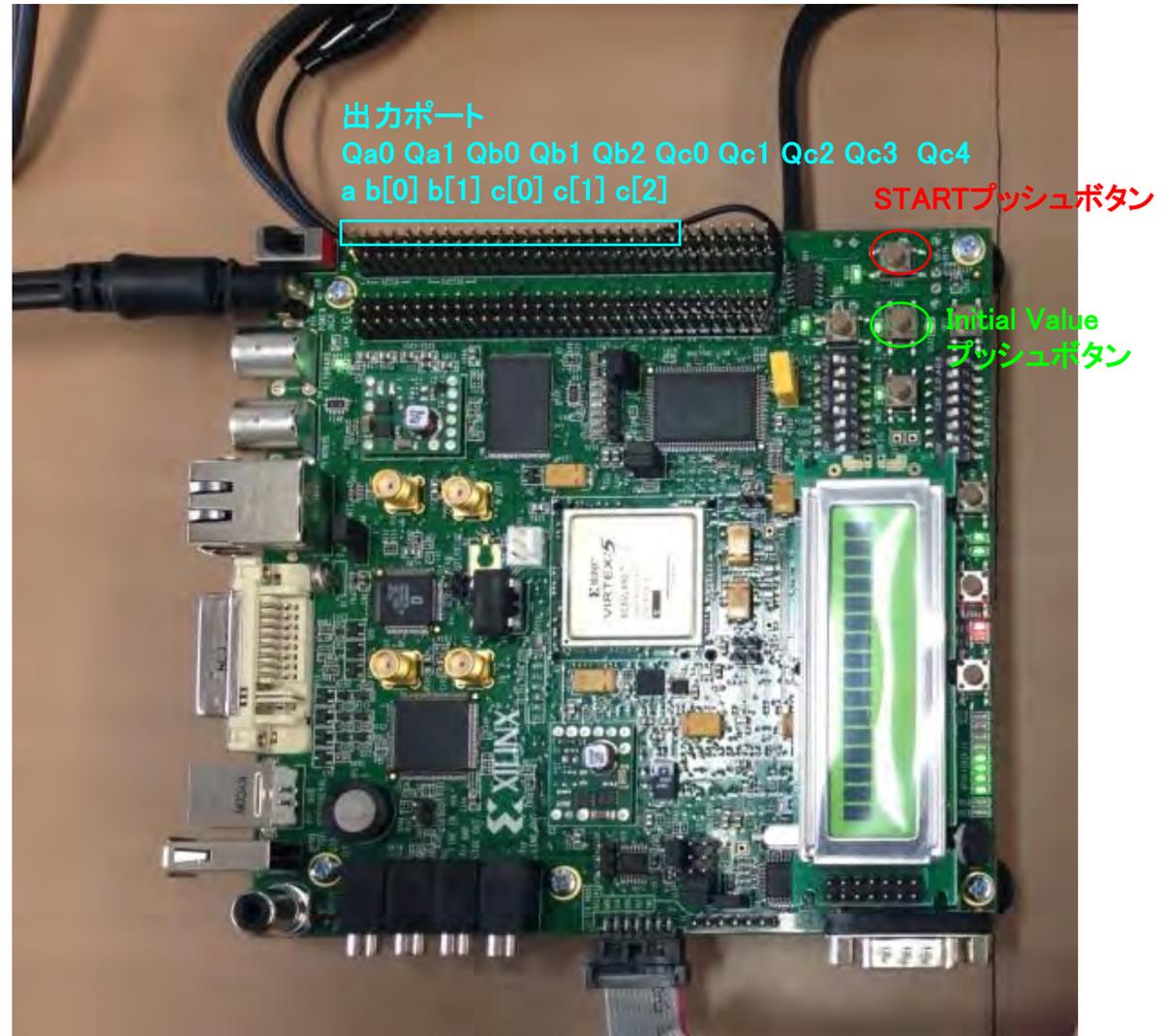
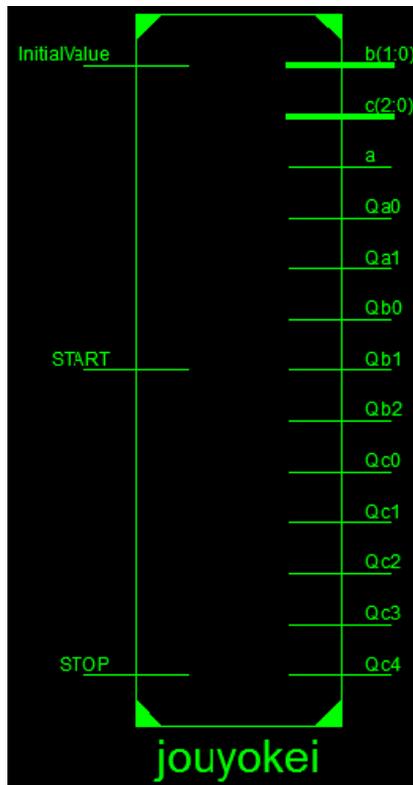
FPGA実装

STOPポートの入力: 100MHz FPGA クロック

Buffer_CLKポートの入力: 33MHz FPGA クロック(バッファの遅延 $\tau = 30.30\text{ns}$)

入力ポート

出力ポート



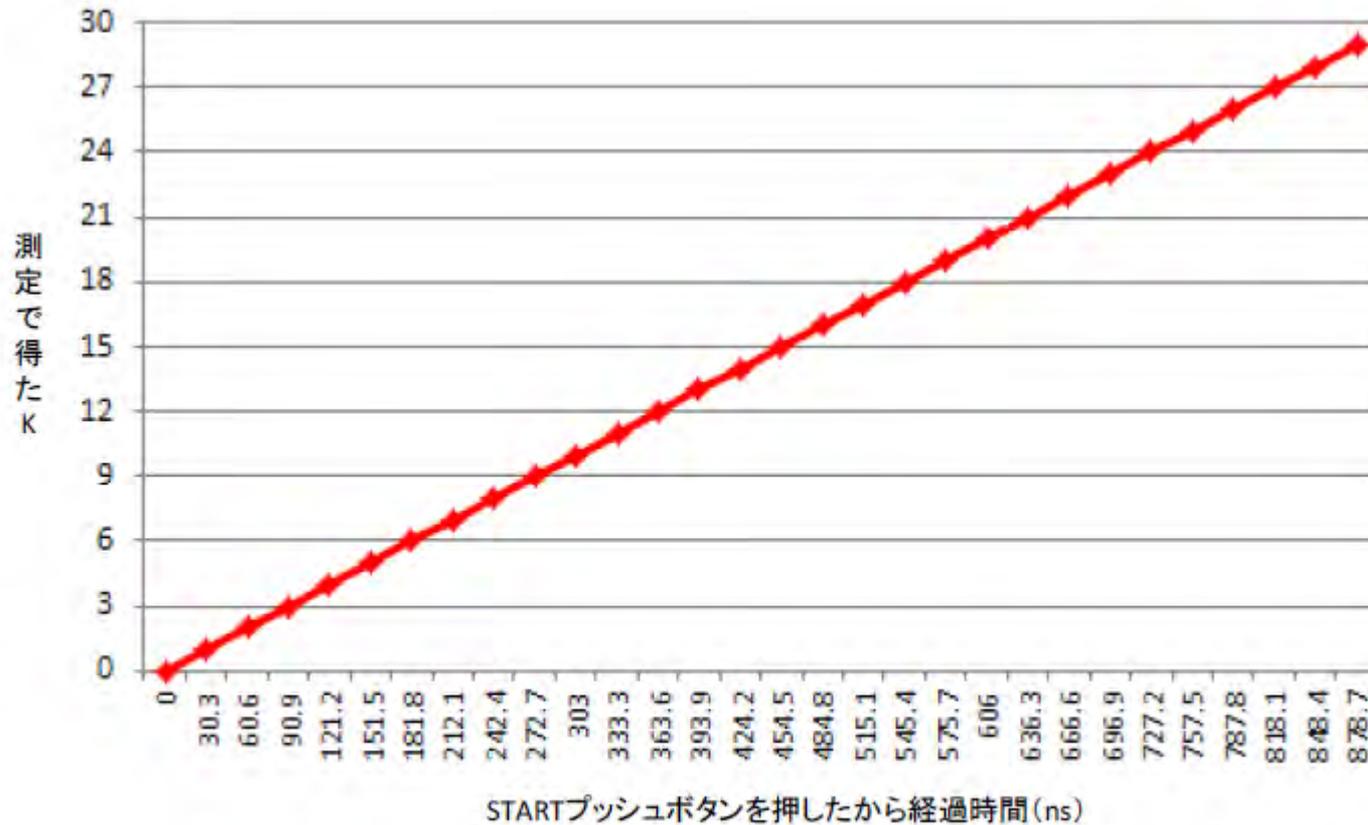
FPGA(Field Programmable Array) 実装

20



FPGA実装 剰余系TDCの評価

剰余系TDC回路はFPGAで実現できることが示された。



経過時間 VS. 測定で得た k

中国人の剰余定理(剰余系)

兵士数を数えるのに使用

群馬大学 阿部優大、小林春夫

Chinese Remainder Theorem



Sun Tzu

Chinese arithmetic book 'Sun Tzu calculation'

孫子算經

“When dividing by 3, its residue is 2,
dividing by 5, its residue is 3,
dividing by 7, its residue is 2.
What is the original number ?”

Answer 23

Generalization



Chinese Remainder Theorem



Sun Tzu calculation

How to use Chinese remainder theorem

He used to quickly find out how many soldiers there are.



Sun Tzu

“Divide into 3.”

Remainder : 2



...



How to use Chinese remainder theorem

He used to quickly find out how many soldiers there are.



Sun Tzu

“Divide into 5.”

Remainder : 3



How to use Chinese remainder theorem

He used to quickly find out how many soldiers there are.



“Divide into 7.”

Sun Tzu



...



Remainder : 2



How to use Chinese remainder theorem

He used to quickly find out how many soldiers there are.



Sun Tzu

“There are 23 people in all according to Chinese remainder theorem”



Example of Residue Number System

$23 \% 3 = 2, \quad 23 \% 5 = 3, \quad 23 \% 7 = 2$

- Natural numbers
3, 5, 7 (relatively prime)
 $N = 3 \times 5 \times 7 = 105$
- k ($0 \leq k \leq N-1 (=104)$)

a : Remainder of k dividing by 3 $a = \text{mod}3(k)$
 b : Remainder of k dividing by 5 $b = \text{mod}5(k)$
 c : Remainder of k dividing by 7 $c = \text{mod}7(k)$



one to one

Chinese remainder theorem

a	b	c	k
0	0	1	15
1	1	2	16
2	2	3	17
0	3	4	18
1	4	5	19
2	0	6	20
0	1	0	21
1	2	1	22
2	3	2	23
0	4	3	24
1	0	4	25
2	1	5	26
0	2	6	27
1	3	0	28
2	4	1	29

Residue number system

解説

孫子算経は兵法家の孫子(孫武)より
ずいぶん前の書で、直接は関係ないようである。

<https://kobaweb.ei.st.gunma-u.ac.jp/news/pdf/2014/2014-07-30joyo.pdf>

が、ここでは孫子が兵士の数を素早く数える
という話にした。



グレイコードを用いた 時間デジタル変換回路

群馬大学大学院 理工学府 電子情報部門

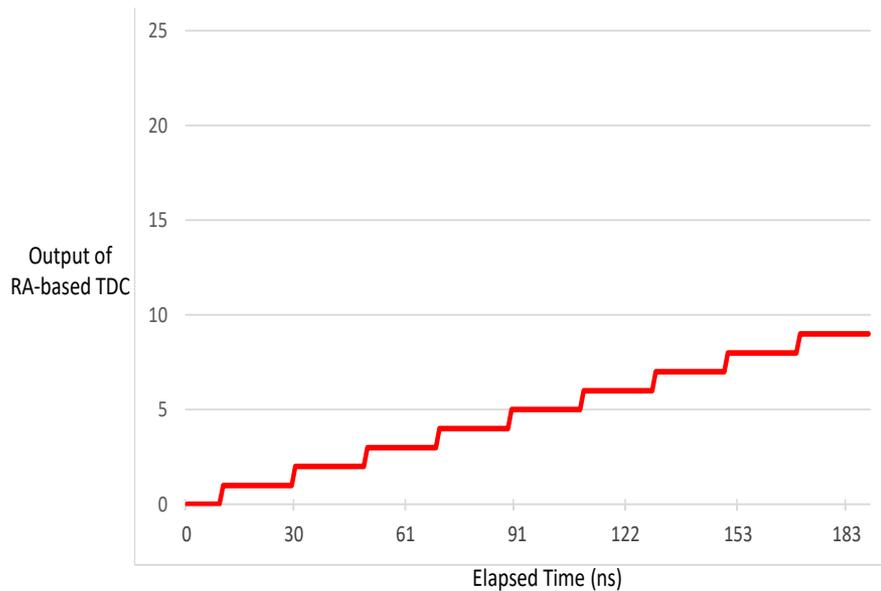
小林春夫

koba@gunma-u.ac.jp

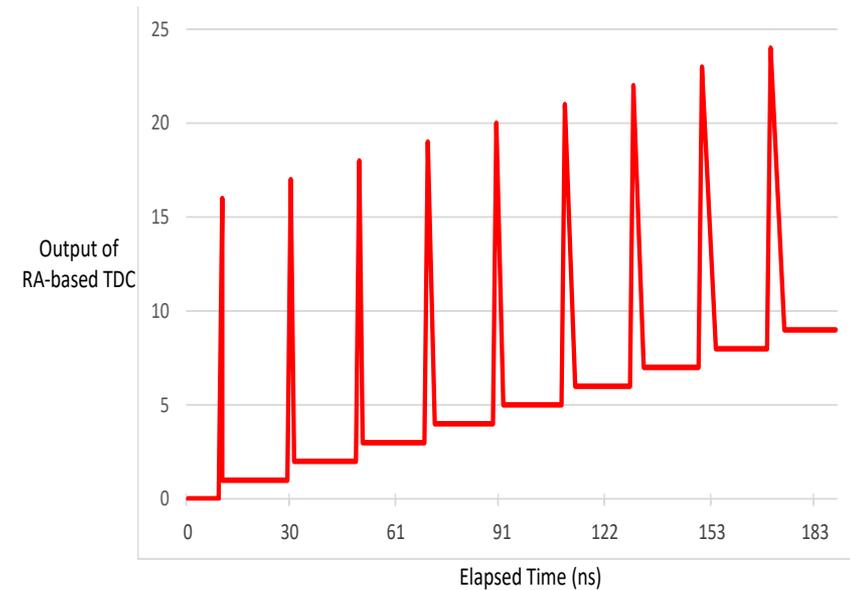
<http://www.el.gunma-u.ac.jp/~kobaweb/>



剰余系TDCと回路非理想特性の影響



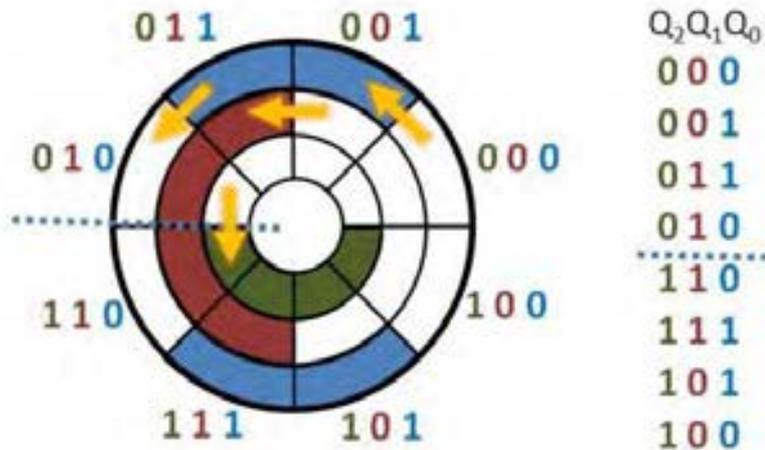
No mismatches among the delay stages



Mismatches exist among the delay stages
(large glitches are observed)

Simulation results with Residue Arithmetic-based TDC without and with mismatches among delay cells in ring oscillators.

グレイコード (Gray code)



Frank Gray and A. L. Johnson in laboratory bench. Behind the glass panels on sides and top are the photo-electric cells.

グレイコード: 前後に隣接する符号間のハミング距離が必ず1

ベル研究所のフランク・グレイが1947年の特許出願書で最初に使用した。

● Gray code の応用

従来例:

AD変換器、ロータリーエンコーダー

群馬大 小林研究室からの提案 (グリッチ低減のため):

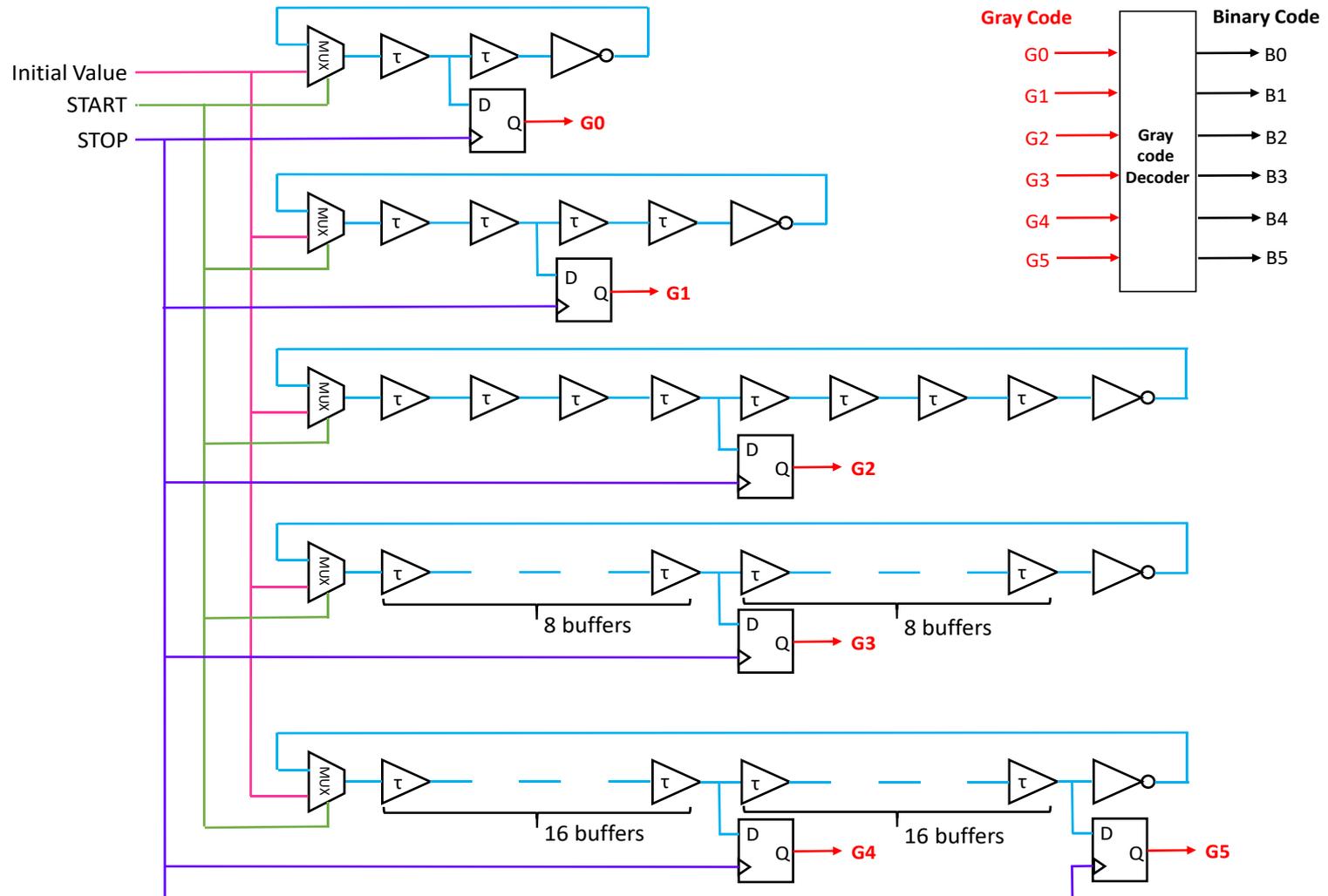
時間デジタイザ回路(TDC)

DA変換器

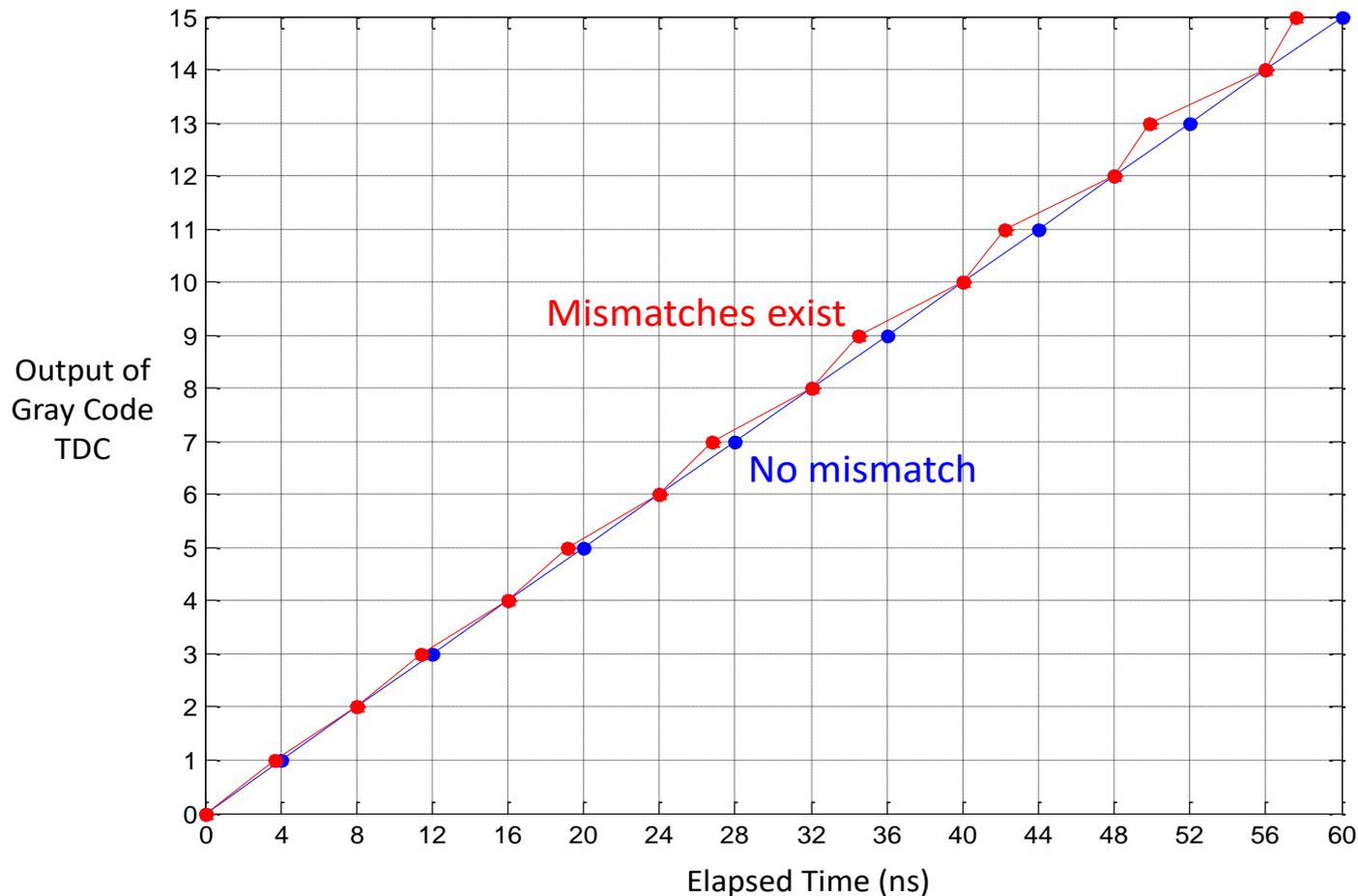
Binary Code & Gray Code

Decimal numbers	Binary Code	Gray Code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Gray code TDC



Gray code TDC と回路非理想特性の影響



RTL simulation results for 4-bit Gray code based TDC without and with one delay mismatch.

ハノイの塔とGray code

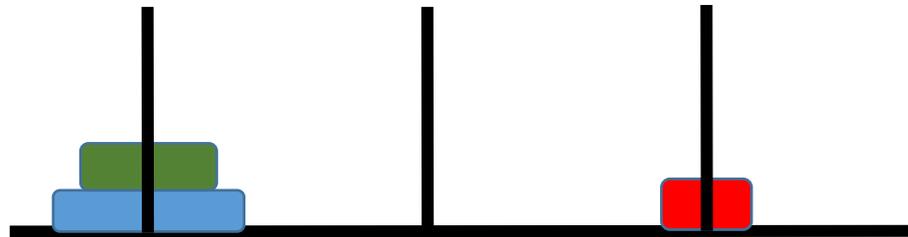
Binary Code と Gray Code

Decimal numbers	Binary Code	Gray Code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

1: 0 0 1

$Q_2 Q_1 Q_0$
0 0 0
0 0 1
0 1 1
0 1 0

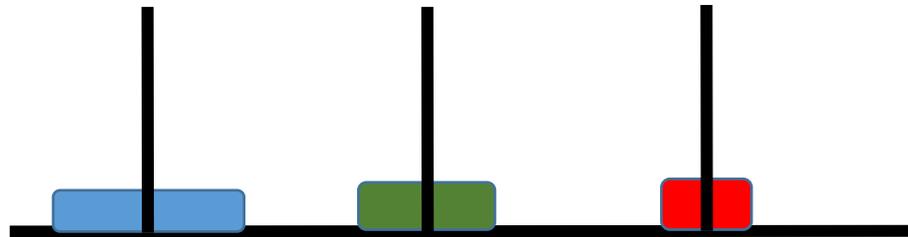
1 1 0
1 1 1
1 0 1
1 0 0



2: 0 1 1

$Q_2 Q_1 Q_0$
0 0 0
0 0 1
0 1 1
0 1 0

1 1 0
1 1 1
1 0 1
1 0 0



3: 0 1 0

 $Q_2 Q_1 Q_0$

0 0 0

0 0 1

0 1 1

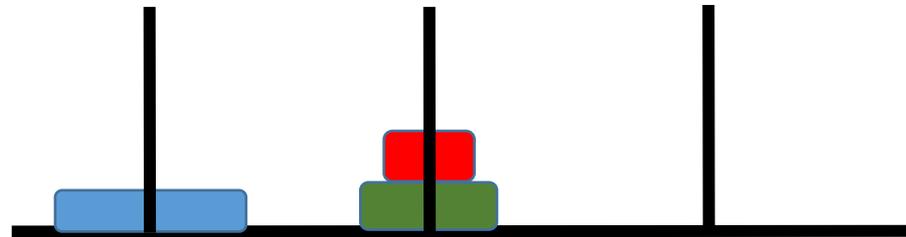
0 1 0

1 1 0

1 1 1

1 0 1

1 0 0



4: 1 1 0

 $Q_2 Q_1 Q_0$

0 0 0

0 0 1

0 1 1

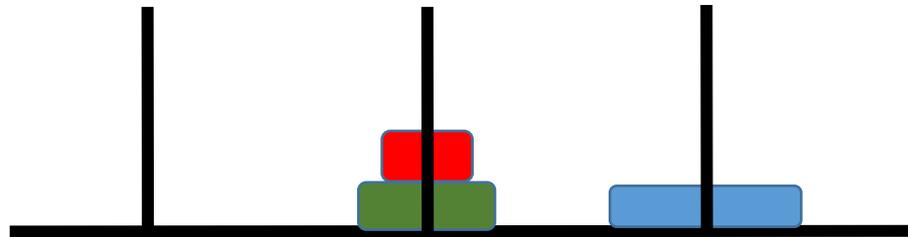
0 1 0

1 1 0

1 1 1

1 0 1

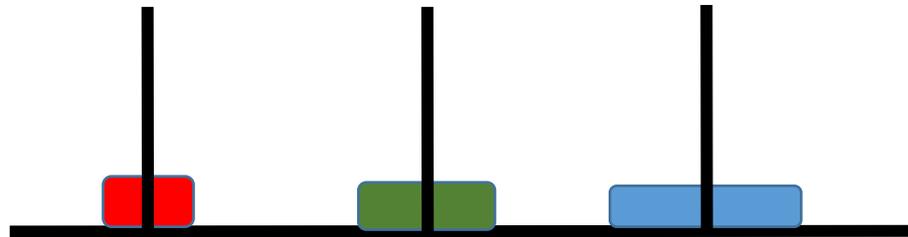
1 0 0



5: 1 1 1

$Q_2 Q_1 Q_0$
0 0 0
0 0 1
0 1 1
0 1 0

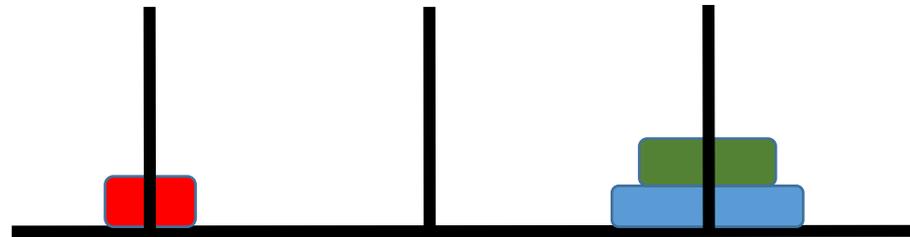
1 1 0
1 1 1
1 0 1
1 0 0



6: 1 0 1

Q_2	Q_1	Q_0
0	0	0
0	0	1
0	1	1
0	1	0

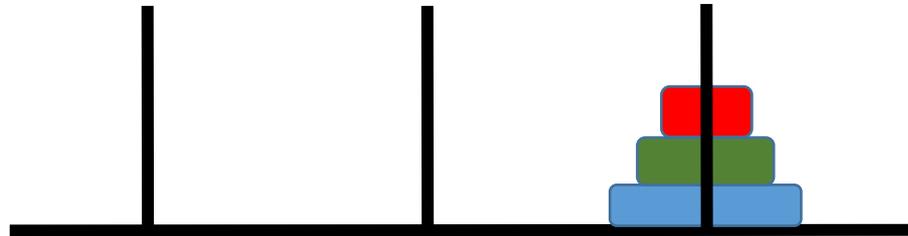
1	1	0
1	1	1
1	0	1
1	0	0



7: 1 0 0

Q_2	Q_1	Q_0
0	0	0
0	0	1
0	1	1
0	1	0

1	1	0
1	1	1
1	0	1
1	0	0



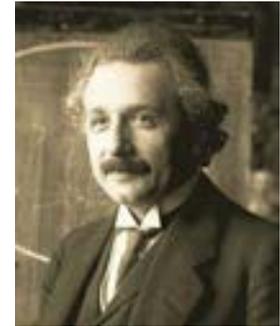
「時間」はミステリアス

往古来今、之を**宙**と謂い
四方上下、之を**宇**と謂う。
淮南子

時間
空間



時空は一体
時間は相対的である。
アインシュタイン



虚数時間
ホーキング博士



時間は最も貴重な資源

「成果を上げる者は、
仕事からスタートしない。
時間からスタートする。

計画からもスタートしない。
まず、何に時間がとられているかを
知ることからスタートする。

次に、時間を奪おうとする非生産的な要求を退ける。
そして、得られた自由な時間を大きくまとめる」



現代技術のベースになる数学と和算の共通性の小さな気付き

－ 電子回路設計と 9 去法, 11 去法

Small Awareness of Commonality Between Modern Technology and Wasan

－ Electronic Circuit Design and Casting Out Nine, Casting Out Eleven

李 雄炎, 桑名 杏奈, 小林 春夫 (群馬大学)

Xiongyan LI, Anna KUWANA, Haruo KOBAYSHI (Gunma University)

Abstract In the digital circuit design using binary representations, the remainder system is often used and it is based on three of the remaining design $2^n - 1, 2^n, 2^n + 1$. Recently we learned about Wasan of casting out nine, casting out eleven in decimal representations, and we inferred from these that it is easy to obtain the residues of $2^n - 1, 2^n, 2^n + 1$ for a number in binary representation; we developed a C program and confirmed their validity with numerical simulation. In other words, we have realized the commonality of the remainder system of digital circuit design in the binary representation and Wasan of casting out nine, casting out eleven in decimal representation.

1. はじめに

筆者らは古典数学をアナログ・デジタル混載の電子回路設計に応用する研究を行っている。例えば魔方陣をデジタル・アナログ変換器の単位回路配列の選択アルゴリズム⁽¹⁾⁽²⁾に用いる、フィボナッチ数列を逐次比較近似アナログ・デジタル変換器の信頼性向上のための冗長設計に用いる⁽³⁾⁽⁴⁾などである。さらに孫子算経（剰余系アルゴリズム）を信号波形の周波数推定に用いる方式⁽⁵⁾⁽⁶⁾や時間デジタル変換回路⁽⁷⁾等のアナログ・デジタル混載回路設計に用いる方式の研究をしてきた。

剰余系アルゴリズムを 2 進数ベースのデジタルプロセッサに用いることは活発に研究されてきている。この分野を長年研究されている群馬大学 魏書剛先生、田中勇樹先生グループの学生さんの発表をたまたま拝聴する機会があったが、 $2^n - 1, 2^n, 2^n + 1$ を基数にした剰余系を用いていた。デジタルではそのようにすることが多いとのことであるがその理由がわからなかった。

一方、2019年10月に集積回路分野の国際学会参加のために中国 重慶市に向かう機上で数学の啓蒙書を読んでいると、古来からインド、中国、さらに和算が盛んであった江戸時代の日本では 9 去法, 11 去法が (10 進数表現の数の) 加減乗算の演算チェックに用いられていることを知った。そのとき 2 進表現での剰余系は $2^n - 1, 2^n, 2^n + 1$ を基数にする理由ではないかと思いつき、数式での解析とプログラム作成をしての検証を行った。その気付きの過程

を紹介する.

2. 10進数表現での9去法と11去法 ⁽⁸⁾

この節では例を用いて9去法, 11去法, および9去法を用いた10進数表現での2つの数の乗算の検算法, および10の剰余を示す.

<2.1> 10進数での9の剰余と9去法

例1. 34671 (10進) の9の剰余は3

$$34671 = 3852 \times 9 + 3$$

9去法: $3+4+6+7+1=21=2 \times 9 + 3$

例2. 29584 (10進) の9の剰余は1

$$29584 = 3287 \times 9 + 1$$

9去法: $2+9+5+8+4 = 28 = 3 \times 9 + 1$

例3. 81275 (10進) の9の剰余は5

$$81275 = 9030 \times 9 + 5$$

9去法: $8+1+2+7+5 = 23 = 2 \times 9 + 5$

10進数表現での数の9の剰余は, 各桁の和の9の剰余と等しい. これが9去法である.

<2.2> 9去法による乗算結果の検算

Aの9の剰余をa, Bの9の剰余をb \rightarrow ABの9の剰余は abの9の剰余になる

例1. $34671 \times 29584 = 1025706864$

$$3 \times 1 = \boxed{3}$$

Aの9の剰余 \times Bの9の剰余

$$1+0+2+5+7+0+6+8+6+4 = 39 = 4 \times 9 + \boxed{3}$$

ABの9の剰余

<2.3> 10進数での11の剰余と11去法

例1. 34671 (10進) の11の剰余は10

$$34671 = 3151 \times 11 + 10$$

11去法: $3-4+6-7+1=-1=-1 \times 11 + 10$

例2. 29584 (10進) の11の剰余は5

$$29584 = 2689 \times 11 + 5$$

11去法: $2-9+5-8+4 = -6 = -1 \times 11 + 5$

例3. 81275 (10進) の11の剰余は7

$$81275 = 7388 \times 11 + 7$$

11去法: $8-1+2-7+5 = 7 = 0 \times 11 + 7$

10進数表現での数の11の剰余は, 各桁を交互に加える引くの演算を行った数の11の剰余と等しい. これが11去法である.

<2.4> 10進数での10の剰余

自明であるが 10 進数表現での最終桁が 10 の剰余である.

<2.5> 数学的な記述

合同式とは, 割り算の余りのみに注目した等式のことである. 例えば, 7 と 4 は, どちらも 3 で割った余りが 1 である. これを, 合同式では $7 \equiv 4 \pmod{3}$ と書く. 「7 合同 4 モッド 3」と読む. 7 と 4 は 3 で割った余りのみに注目すれば等しいという意味である. より一般に, a と b を n で割った余りが等しいとき, 合同式では $a \equiv b \pmod{n}$ と書く. 合同式の応用として, 大きな数を 9 や 11 で割ったときの余りを簡単に求める方法がある.

9 去法

9 去法は合同式 $10 \equiv 1, 10^2 \equiv 1, 10^3 \equiv 1, 10^4 \equiv 1, 9 \equiv 0 \pmod{9}$ が成り立つことを利用して, ある整数を 9 で割ったときの余りを求める方法である. 例として 2300 を 9 で割ったときの余りを求める. $10^3 \equiv 1, 2 \equiv 2 \pmod{9}, 10^2 \equiv 1, 3 \equiv 3 \pmod{9}$ なので, 合同式の乗算定理より $2 \times 10^3 \equiv 1 \times 2 \pmod{9}, 3 \times 10^2 \equiv 1 \times 3 \pmod{9}$ が成立する. 合同式の加算定理より, $2 \times 10^3 + 3 \times 10^2 \equiv 2 + 3 \equiv 5 \pmod{9}$ となり, 余りは 5 であることがわかる.

11 去法

11 去法の理屈と 9 去法は同じである. 例として 52364 を考える. 11 去法は合同式 $1 \equiv 1, 10 \equiv -1, 10^2 \equiv 1, 10^3 \equiv -1, \dots \pmod{11}$ が成り立つことを用いて, 整数を 52364 で割ったときの余りを求める方法である. 52364 を 11 で割ったときの余りは $52364 \equiv 4 \times 1 + 6 \times (-1) + 3 \times 1 + 2 \times (-1) + 5 \times 1 \equiv 4 \pmod{11}$ となる.

3. 2 進数表現での剰余

前節での 10 進数表現での 9 去法, 11 去法にヒントを得て, 2 進数表現での $2^n - 1, 2^n + 1$ の剰余が簡単に求まるのではないかと類推し, 確認した. 2^n の剰余は自明に求まる.

例: $n=4$ の場合

1010 1111 0111 (2 進数表現)

A F 7 (16 進数表現)

$10 \times 16 \times 16 + 15 \times 16 + 7 = 2807$ (10 進数表現)

① この数の $2^4 - 1 = 15$ による剰余は 2

$$2807 = 187 \times 15 + 2$$

各桁の和(9 去法の類推):

$$A + F + 7 \text{ (16 進)} = 10 + 15 + 7 \text{ (10 進)} = 32 \text{ (10 進)} = 2 \times 15 + 2 \text{ (10 進)}$$

② $2^4 + 1 = 17$ の剰余は 2

$$2807 = 165 \times 17 + 2$$

各桁の和差 (11 去法の類推):

$$A - F + 7 \text{ (16 進)} = 10 - 15 + 7 \text{ (10 進)} = 2 \text{ (10 進)} = 0 \times 17 + 2 \text{ (10 進)}$$

③ 2^4 の剰余は 7

A F 7 (16進数表現)

より数学的な記述は以下のようなになる.

8進表現での「9去法と11去法」との対応の考察

10進表現での9去法と11去法より,8進の場合の「9去法と11去法」に対応するものはどうなるかを考察した.8進の場合で,同じく+1と-1をすれば,剰余も簡単に求められると類推した.8進の場合,10進の9去法に対応するものは $2^3-1=7$ 去法になる.8進表現で4200を例として7で割ったときの余りを求める. $4\equiv 4(\text{mod}7)$, $2\equiv 2(\text{mod}7)$, $0\equiv 0(\text{mod}7)$, $4200\equiv 4+2\equiv 6(\text{mod}7)$ となつて,4200の余りが6であることがわかる.

8進の場合で,「10進の11去法」は $2^3+1=9$ 去法になる.8進表現4200を例として10進表現9で割ったときの余りを求める. $4\equiv 4(\text{mod}9)$, $3\equiv 3(\text{mod}9)$, $2\equiv 2(\text{mod}9)$, $1\equiv 1(\text{mod}9)$, $4200\equiv 4+2\equiv 2\equiv 7(\text{mod}11)$ となつて,4200の余りが7であることがわかる.

16進表現での9去法と11去法

16進の場合で,9去法は $2^4-1=F$ (10進に換算しては15)去法にして求める.まずは16進の4200と9151を例としてF(10進に換算しては15)で割ったときの余りを求める. $16^3\equiv 1$, $4\equiv 4$, $9\equiv 9(\text{mod}F)$, $16^2\equiv 1$, $2\equiv 2$, $5\equiv 5(\text{mod}F)$, $F\equiv 0(\text{mod}F)$, $4200\equiv 4+2\equiv 6(\text{mod}F)$, $9151\equiv 9+1+5+1\equiv F\equiv 0(\text{mod}F)$ となつて,4200を9151の余りが6と0であることがわかる.

16進の場合で,11去法は $2^4+1=11$ (10進に換算しては17)去法にして求める.16進表現の4200と9151を例として11(10進に換算しては17)で割ったときの余りを求める. $16^3\equiv -1$, $4\equiv 4$, $9\equiv 9(\text{mod}F)$, $16^2\equiv 1$, $2\equiv 2$, $5\equiv 5(\text{mod}F)$, $F\equiv 0(\text{mod}F)$, $4200\equiv 4+2\equiv -2\equiv E(\text{mod}F)$, $9151\equiv 9+1-5+1\equiv 4(\text{mod}F)$ となつて,4200を9151の余りがEと4であることがわかる.

 2^n 進表現での9去法と11去法

10進,8進と16進の結論で得られたのと同じように, 2^n 進の場合は「10進での9去法」が 2^n-1 去法にして整数の余りも簡単に求められる.同じく「10進での11去法」は 2^n+1 去法にして整数の余りを簡単に求められる.このとき 2^n-1 去法は $2^{nm}\equiv 1 \pmod{2^n-1}$ を利用する.例えば,整数 abc を 2^n-1 で割ったときの余りは $abc\equiv a+b+c \pmod{2^n-1}$ となり,余りが $a+b+c$ であることがわかる.

2^n+1 去法は $2^{nm}\equiv 1 \pmod{2^n+1}$ (m は偶数), $2^{nm}\equiv -1 \pmod{2^n+1}$ (m は奇数,1を除く)を利用する.例えば,整数 abc を 2^n+1 で割ったの余りは $abc\equiv a-b+c \pmod{2^n+1}$ となり,余りが $a-b+c$ であることがわかる.また,これらの剰余系はメルセンヌ数 2^n-1 ,フェルマー数 2^n+1 の形であることにも気が付く.(メルセンヌ数は巨大素数探索にも利用され,4月現在,発見されている最大の素数は $2^{82589933}-1$ で,2486万2048桁ある.)

4. 古代中国の数学書

剰余系で用いる「中国の剰余定理」は,中国の算術書『孫子算経(そんしさんけい)』に由来する整数の剰余に関する定理である.『孫子算経』には,日本で最も多く利用された和算

学習書『塵劫記』の中の“百五減算”で知られる「3で割ると2余り,5で割ると3余り,7で割ると2余る数は何か」という問題とその解法が書かれている. 答えは23である. 中国の剰余定理の最も基本的な形は次のような形式で述べることができる. 与えられた二つの整数 m, n が互いに素ならば, 任意に与えられる整数 a, b に対し, 連立合同方程式 $x \equiv a \pmod{m}, x \equiv b \pmod{n}$ を満たす整数 x が mn を法として一意的に存在する.

『孫子算経』は一千五百年前, 中国の南北朝時代に書かれた算術書であり, 唐代に編纂された算経十書の1つである. この著者の「孫子」について詳細はよくわかっていない. 日本でも知られた『孫子兵法』の作者である孫武より時代はずいぶん前であるので孫武は『孫子算経』の著者ではない. 孫武の子孫とされる孫臏の『孫臏兵法』も中国では有名であり, 孫臏も「孫子」と呼ばれる.

『周髀算経(しゅうひさんけい)』, 『九章算術(きゅうしょうさんじゅつ)』, 『五曹算経(ごそうさんけい)』, 『夏侯陽算経(かこうようさんけい)』, 『五經算術(ごけいさんじゅつ)』, 『緝古算経(しゅうこさんけい)』, 『張邱建算経(ちょうきゅうけんさんけい)』, 『綴術(てつじゅつ)』, 『海島算経(かいとうさんけい)』と『孫子算経』合わせて『算経十書』と称されている. 『算経十書』は古代中国で唐代の官僚や庶民の子弟を集めて数学を教える学校(「算学」といった)で教科書として使用された. その内容は全て古漢語で書かれているので, 中国人でもすごく読み辛く, 現在の中国の小中高の数学教育等にもほとんど使われていない. 日本では『九章算術』が奈良時代から算博士官吏登用試験で利用されていた.

また, 『孫子算経』には剰余系以外にもさまざまな問題が記述されている. その中で雉兔同籠(じとどうりゅう)は非常に面白い. これは「同じ籠の中に雉と兎がいる. 頭の数は35で, 足の数は94である. 雉, 兎それぞれの数は幾らか」の問題である. これが日本に伝わり, 1810年に出版された坂部広胖の著書「算法点竄指南録(さんぼうてんざんしなんろく)」では, 「雉兔」が「鶴亀」に変化する. 「鶴亀合百頭あり, 只云足数和して二百七十二, 鶴亀各何ほどと問」. これが日本の鶴亀算の原点である. このように『孫子算経』は「和算」に影響を与えている.

5. まとめ

10進数の9去法, 11去法と, 2進数表現を用いた剰余系でのデジタル電子回路設計で $2^n - 1, 2^n, 2^n + 1$ の3つを基数とした剰余系システムの共通性に気がついた. この分野の専門家の群馬大学 魏書剛先生に問い合わせたところ次のことである.

「 r 進数の表現を用いた場合, r^n あるいは $r^n - 1, r^n + 1$ を法とすることにより, 除算による余りを求める剰余演算の実現は簡単になります. デジタル回路は2進数系を用いた算術演算の実現手段なので剰余演算も2進数を用いて実現することは基本的な考え方です.

$2^n - 1, 2^n, 2^n + 1$ を法とする剰余演算は, デジタルハードウェア上, 簡単に実現されます.」和算の9去法, 11去法がきっかけになりこのことを知ることができた. 筆者らは整数の面白い性質を積極的に利用するアナログ・デジタル混載電子回路設計の研究を行ってきているが, 今ま

での筆者らの研究では、剰余系の利用はデジタルシステムの場合とは異なりこの性質は使っていない。これを積極的に応用することを考えていきたい。これからも和算や古代中国・インドの数学にアナログ・デジタル混載電子回路設計にアイデアを求めていきたい。

謝辞： 有意義なコメントをいただきました田部井勝稲先生に感謝します。

【参考文献】

- [1] 東野将史, 小林春夫「魔方陣レイアウトによる DA 変換器の直線性の改善」
和算ジャーナル 第 2 号 (会報通巻 52), (2018 年 3 月)
- [2] M. Higashino, S. N. B. Mohyar, Y. Dan, Y. Sun, A. Kuwana, H. Kobayashi, "Digital-to-Analog Converter Layout Technique and Unit Cell Sorting Algorithm for Linearity Improvement Based on Magic Square", Journal of Technology and Social Science (Jan. 2020).
- [3] 小林春夫, 小林佑太郎「フィボナッチ数列と電子回路の話し」
群馬県和算研究会会報 第 50 号 (2016 年 3 月)
- [4] Y. Kobayashi, H. Kobayashi, "Redundant SAR ADC Algorithm Based on Fibonacci Sequence", Advanced Micro-Device Engineering VI, Key Engineering Materials (2016).
- [5] 阿部優大, 片山翔吾, 李 从兵, 小林春夫「孫子算経 (剰余系) の電子回路設計への応用」
第 14 回全国和算研究大会, 栃木県佐野市 (2018 年 8 月 26 日)
- [6] Y. Abe, S. Katayama, C. Li, A. Kuwana, H. Kobayashi, "Frequency Estimation Sampling Circuit Using Analog Hilbert Filter and Residue Number System",
13th IEEE International Conference on ASIC, Chongqing, China (Oct. 2019).
- [7] C. Li, K. Katoh, H. Kobayashi, J. Wang, S. Wu, S. N. B. Mohyar,
"Time-to-Digital Converter Architecture with Residue Arithmetic and its FPGA Implementation",
11th International SoC Design Conference, Jeju, Korea (Nov. 2014).
- [8] Excel VBA 数学教室 (9 去法と 11 去法) <https://excelmath.atelierkobato.com/9kyohou/>

付録： 研究室学生の阿部優大君が 2018 年 8 月に栃木県佐野市で開催された第 14 回全国和算研究大会で剰余系を用いた波形サンプリング回路方式を発表させてもらったが⁽⁵⁾, さらに 2019 年 10 月に中国の重慶市で開催された米国電気学会 (IEEE) 主催国際会議で発表し⁽⁶⁾, Excellent Student Paper Award を受賞した。 (写真)



集積回路システム工学
講義資料

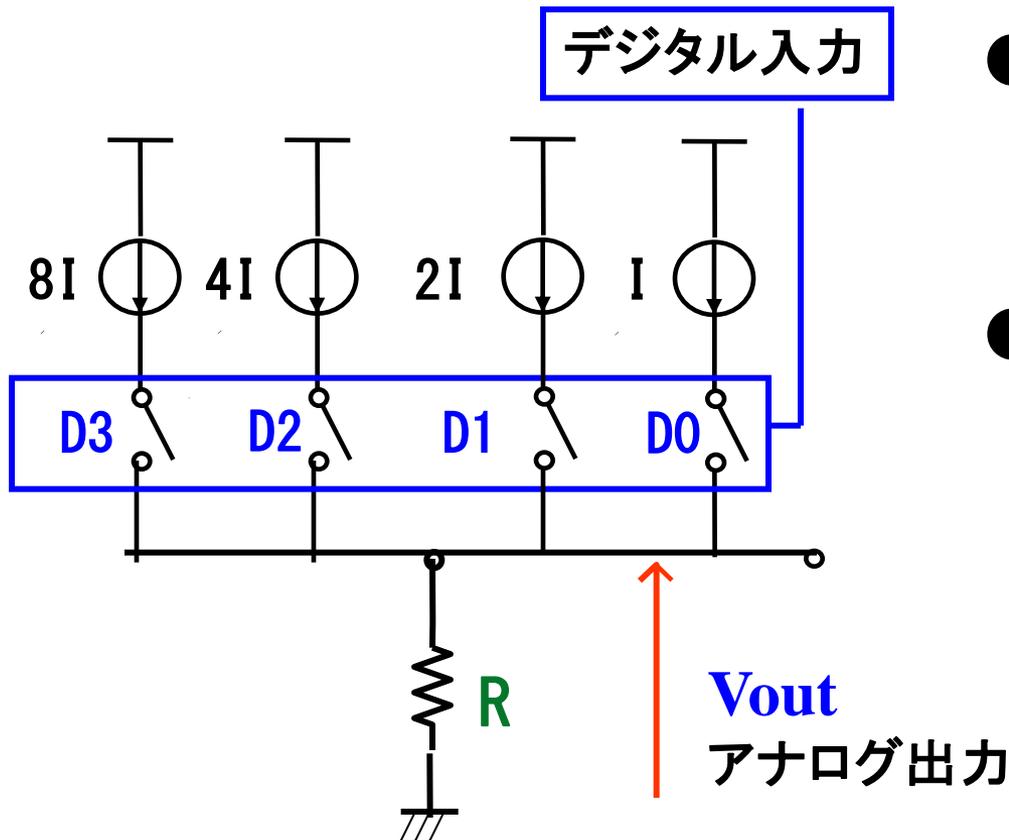
AD/DA 変換器の失敗事例

教科書にでているのはうまくいった技術のみ

群馬大学 小林春夫



電流型2進重み付けDA変換回路 (回路)



● メリット

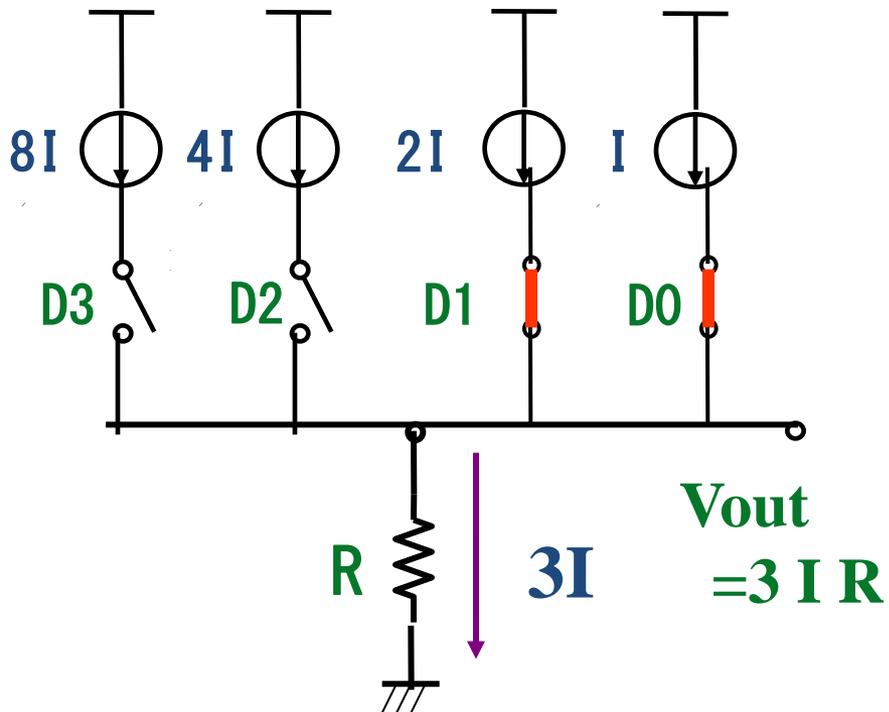
- ・回路規模が小さい
- ・サンプリング速度が速い

● デメリット

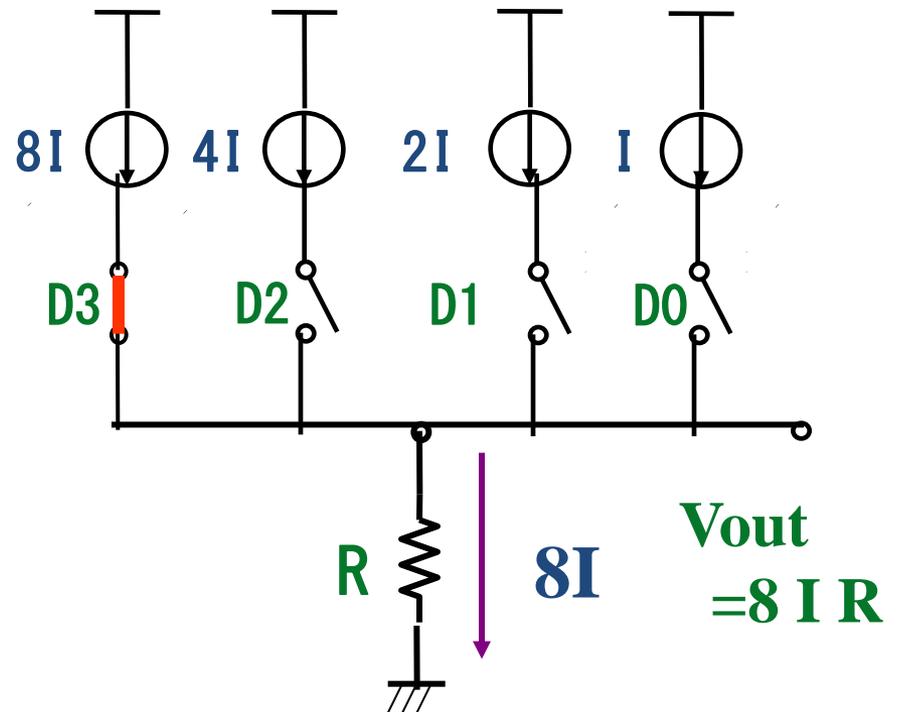
- ・グリッチが大きい
- ・入出力間の単調性が確保出来ない

電流型2進重み付けDA変換回路 (動作)

例: 入力データが3のとき



例: 入力データが8のとき



電流型2進重み付けDA変換回路 (原理)



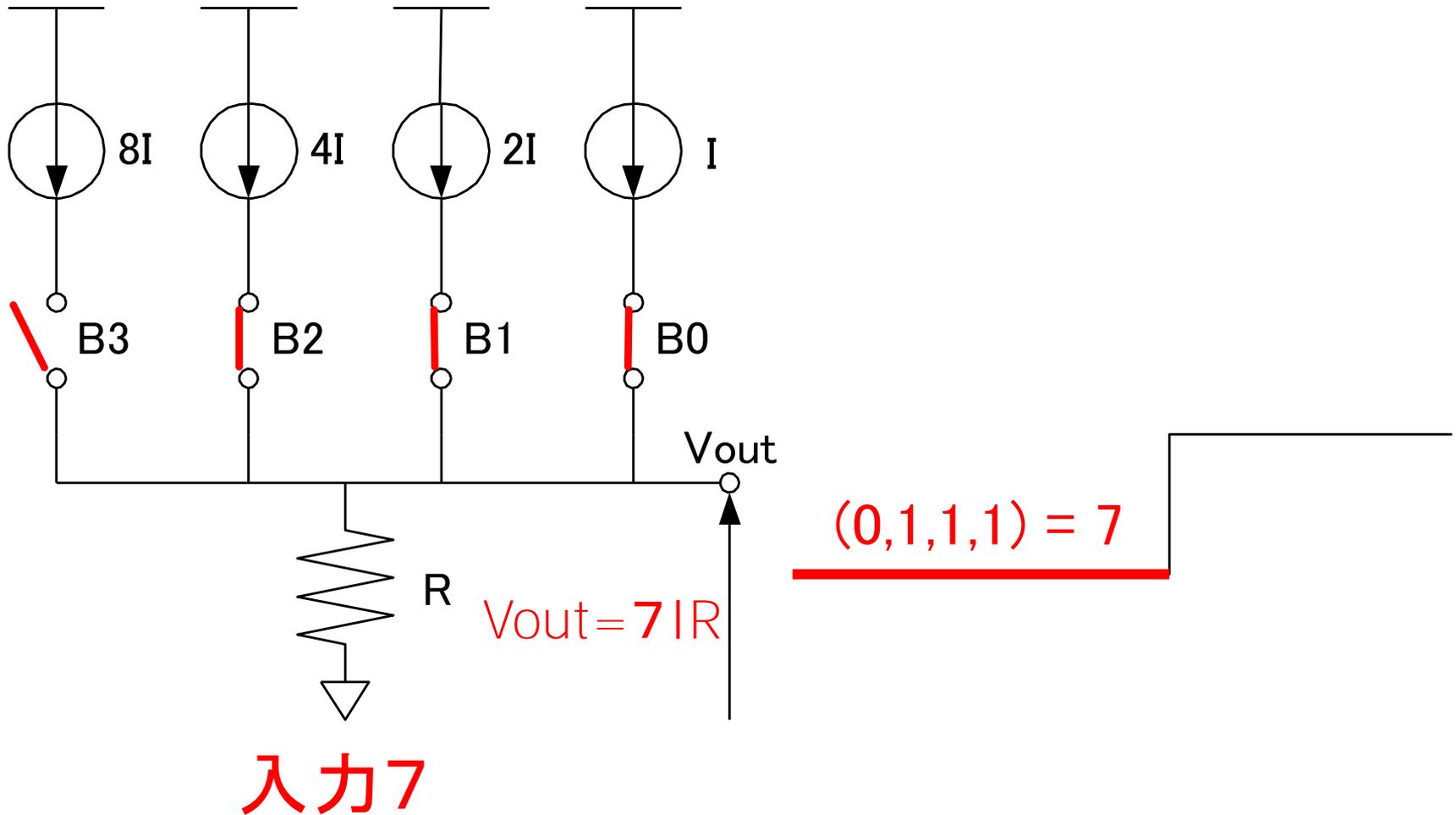
デジタル 入力データ	スイッチ				出力
	D3	D2	D1	D0	Vout
0	0	0	0	0	0
1	0	0	0	1	1R
2	0	0	1	0	2R
3	0	0	1	1	3R
4	0	1	0	0	4R
5	0	1	0	1	5R
6	0	1	1	0	6R
7	0	1	1	1	7R
8	1	0	0	0	8R
⋮			⋮		⋮
15	1	1	1	1	15R

スイッチ 1 のとき ON
0 のとき OFF

デジタル入力データに
比例したアナログ出力
Vout が生成される。

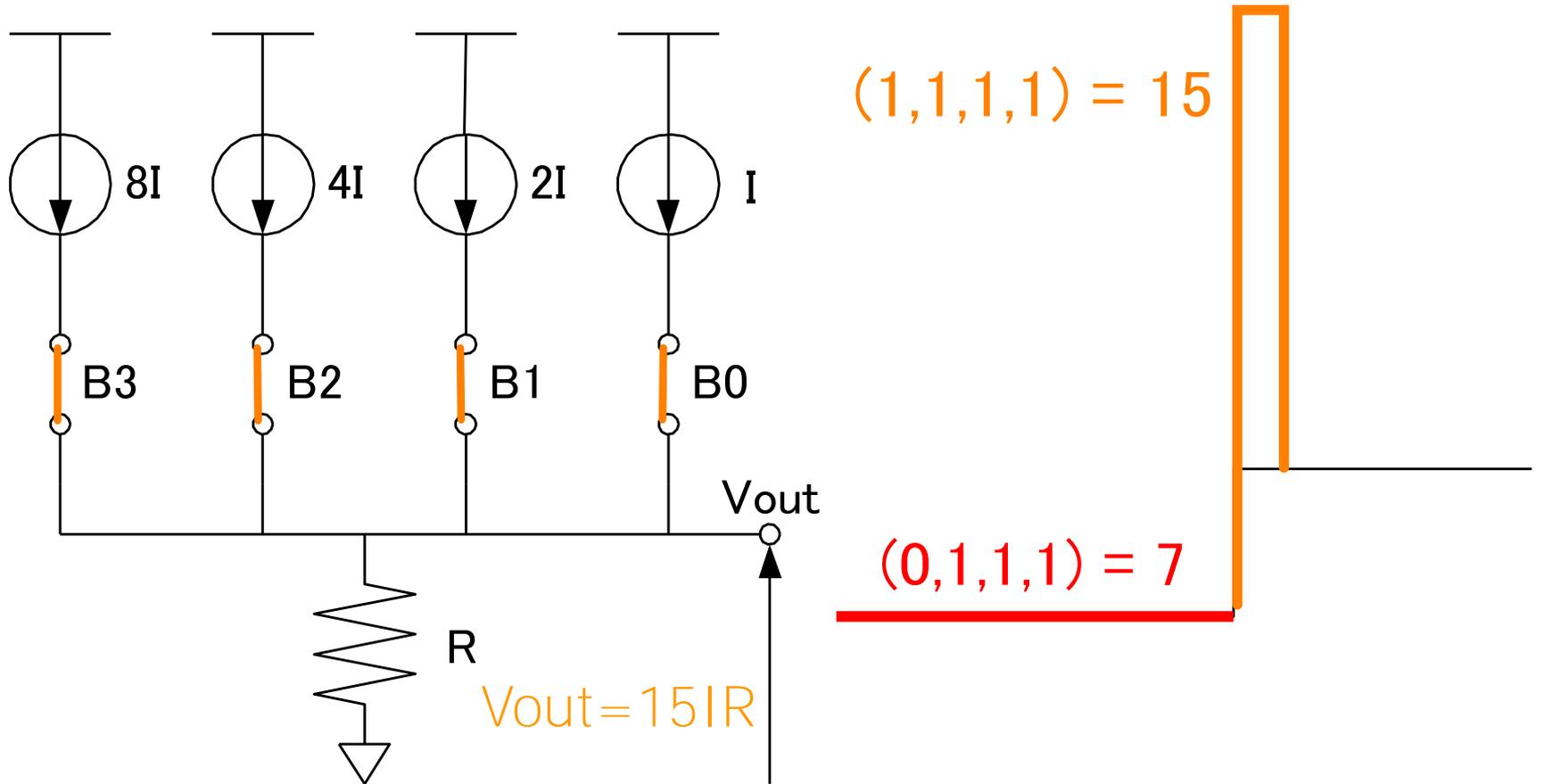
スイッチ切り替えタイミング

スキューが有る場合



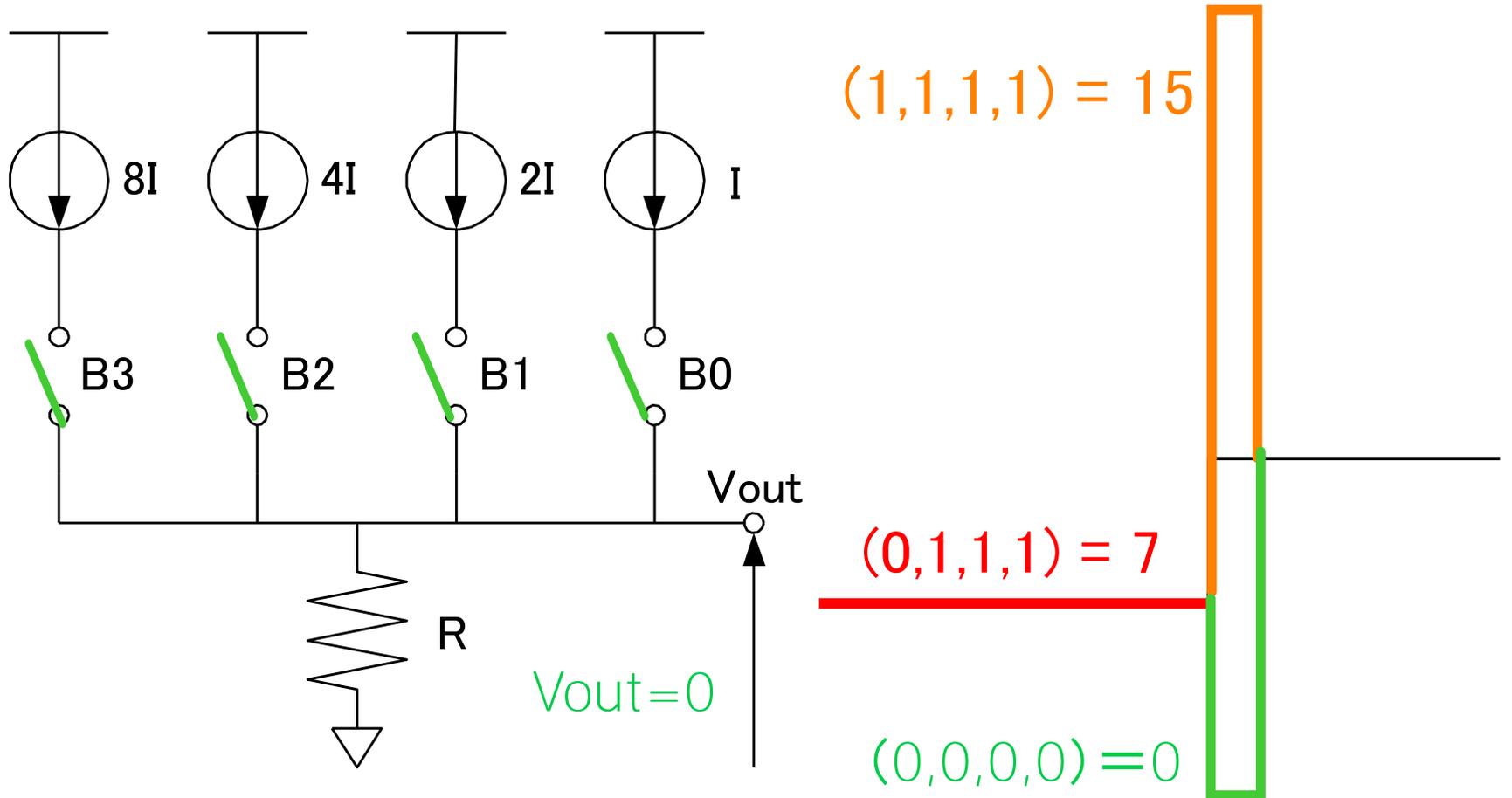
スイッチ切り替えタイミング

スキューが有る場合



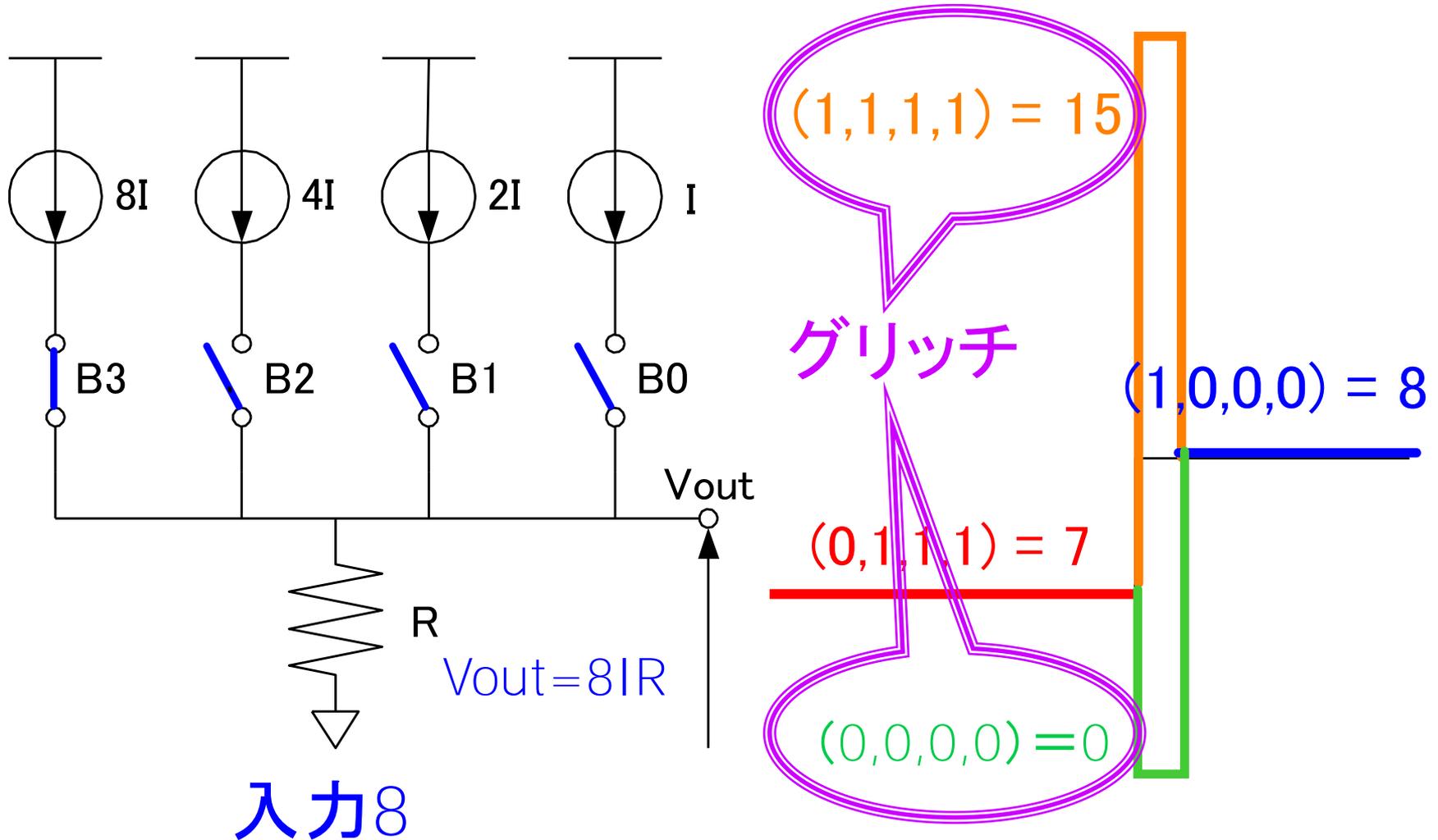
スイッチ切り替えタイミング

スキューが有る場合



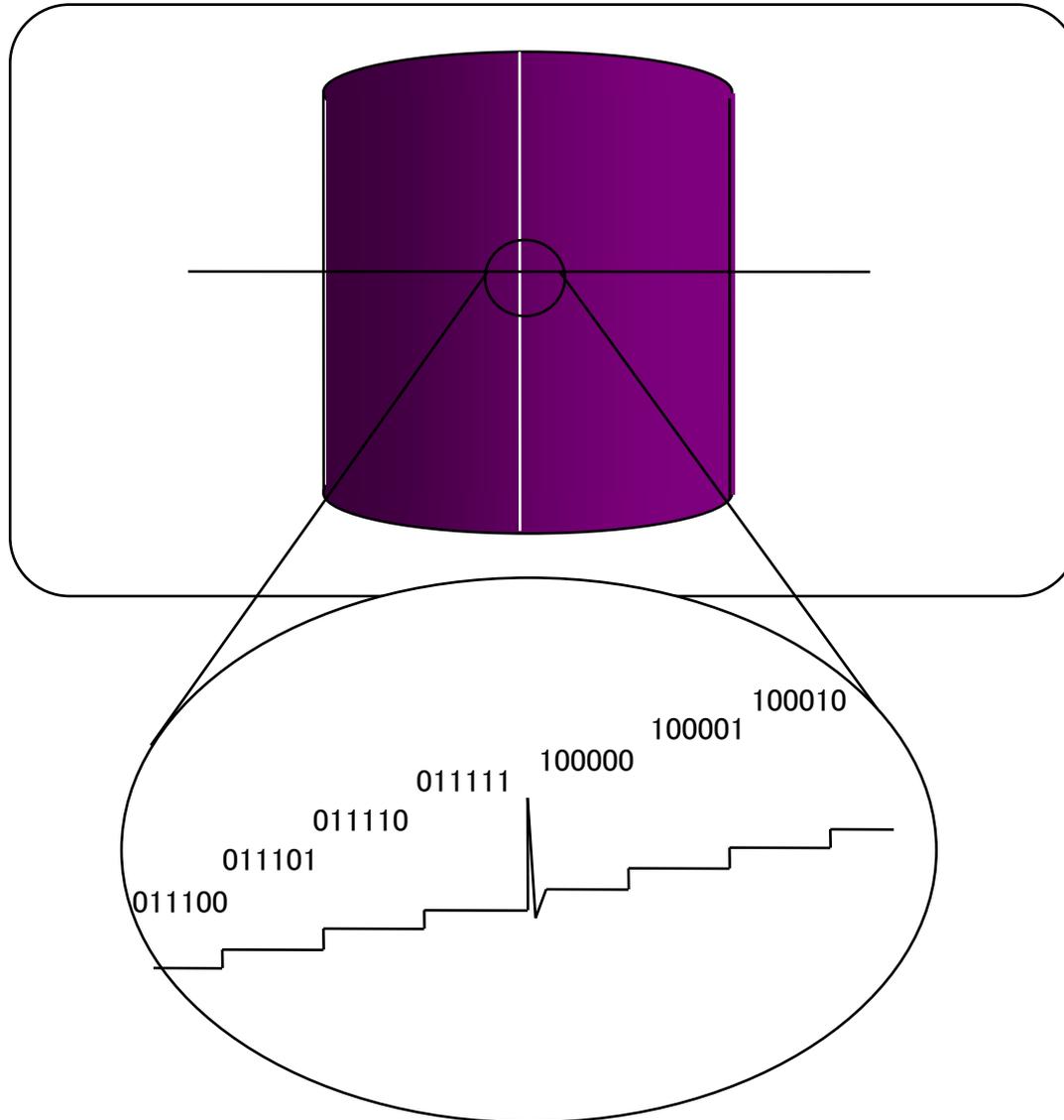
スイッチ切り替えタイミング

スキューが有る場合



DA変換器におけるグリッチの影響

グラフィックディスプレイ用



人間の視覚は
グリッチに敏感

Binary code と Gray code

Decimal numbers	Binary Code	Gray Code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Gray code の特長

対応する
decimal numberが
プラス1
または
マイナス1



1つのビットのみ反転

Gray code とAD変換器

Gray code は ロバスト(頑健)



AD変換器は 信号を
アナログからデジタルの世界へ
(不連続な領域間の遷移)



AD変換器では
まず Gray codeに変換し
その後 Binary codeに変換する構成も多い

Gray code 構成のDA変換器は？

AD/DA変換器の研究をはじめたときに

Gray code 構成のDA変換器が実現できれば



グリッチがなくなる。

回路の教科書/論文のどこにも

Gray code 構成のDA変換器は記述されていない。



革新的構成か？

専門家に聞く

Asad Abidi 先生 (UCLA)

「Gray code 構成でDA変換器が実現できないか
多くの研究者が考えたが、
できないということが(経験的に)わかった。」



教科書にはこのことは書いていない。
学会等でも誰も言わない。
このことから多くのことを学ぶ。

経験論: 観察や実験などを繰り返し行うことによって、
経験を少しずつ積み、結果的に真理に到達する
フランシス・ベーコン

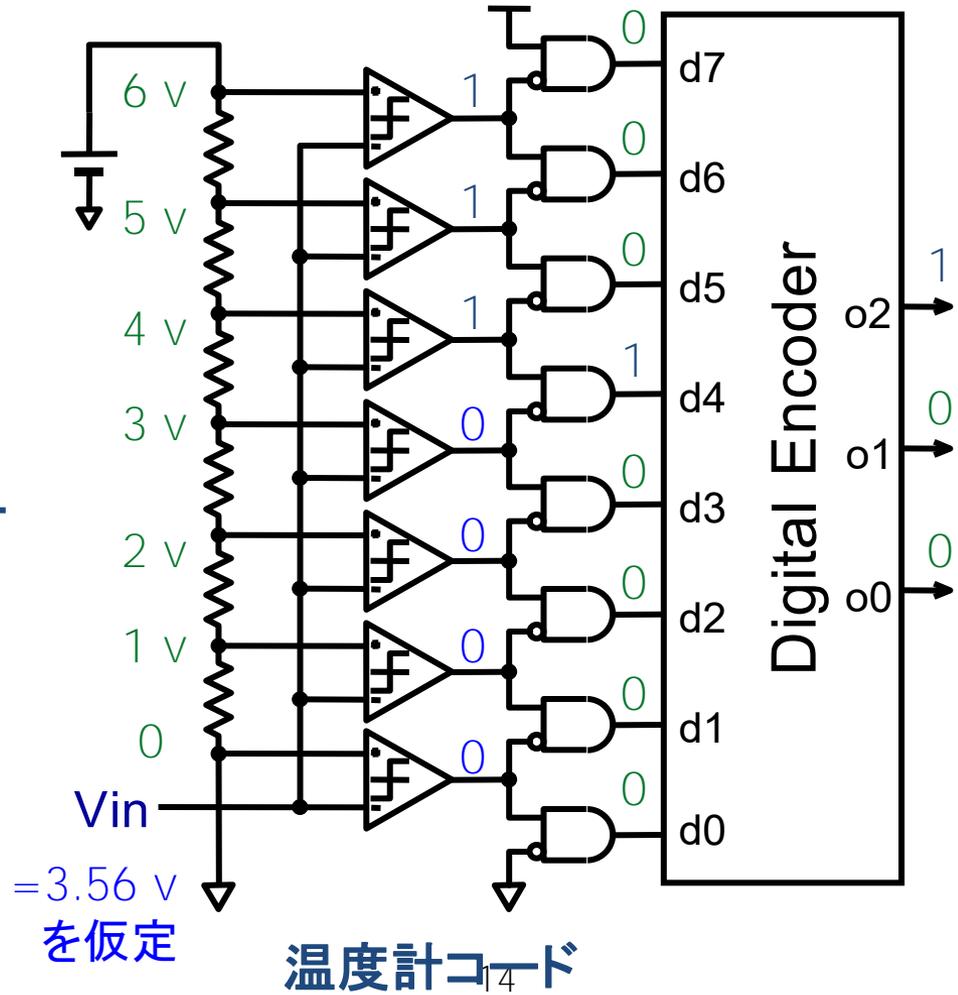
フラッシュ型ADC

長所 : 高速

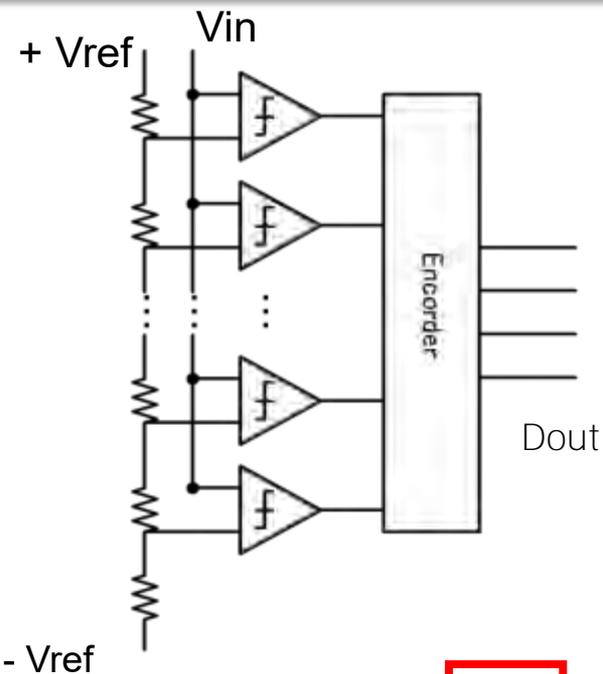
短所 : 回路量 大
消費電力 大
入力容量 大

Encoder 真理値表

d7	d6	d5	d4	d3	d2	d1	d0	o2	o1	o0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

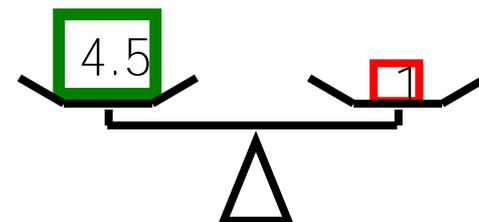
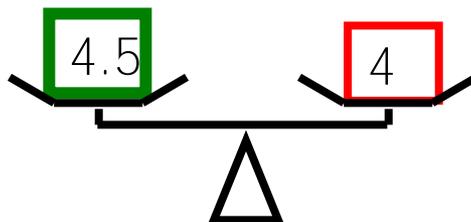
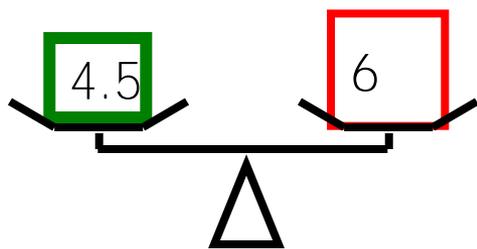
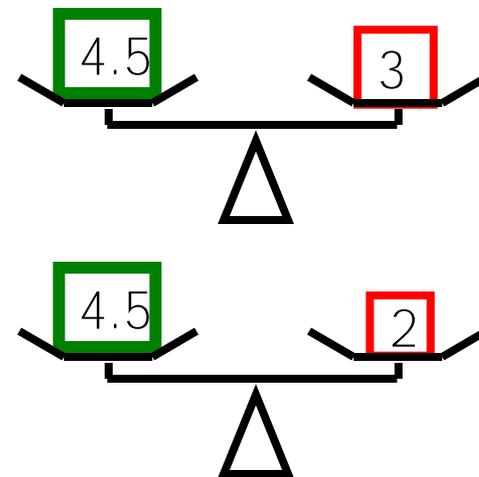
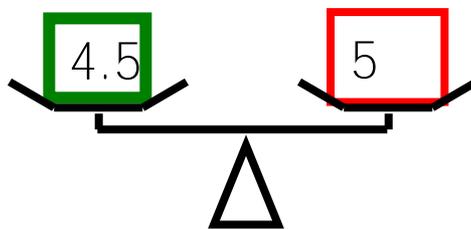
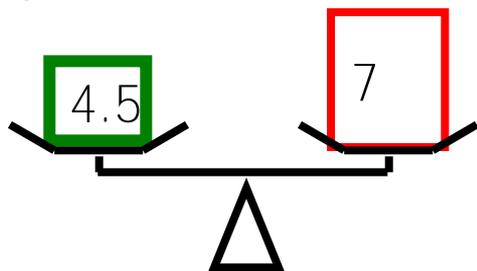


フラッシュ型ADC 大きな冗長性



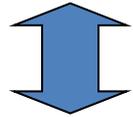
全ての重さの分銅と
それを載せる天秤を用意

入力Vin 4.5



フラッシュ型ADCへの見方

「フラッシュ型ADCは無駄な回路が多く賢い構成ではない」
「6bit フラッシュADC など目をつぶっても実現できる」



「フラッシュ型ADCは偉大な構成」

- 低分解能・超高速ADCのアーキテクチャとしてフラッシュ型を超えようとして、
(公表されてないが、まわりで)
いくつもの研究が失敗している
(UCLA Abidi 先生)
- 産業界で フラッシュ型は生き残っている。

定説へのささやかな挑戦

が、本当に Gray Code 入力DACは
実現できないのか？

やってみたら 出来た！

Gray-code入力DA変換器の検討

群馬大学理工学府 修士2年
姜 日晨* Gopal Adhikari 小林 春夫

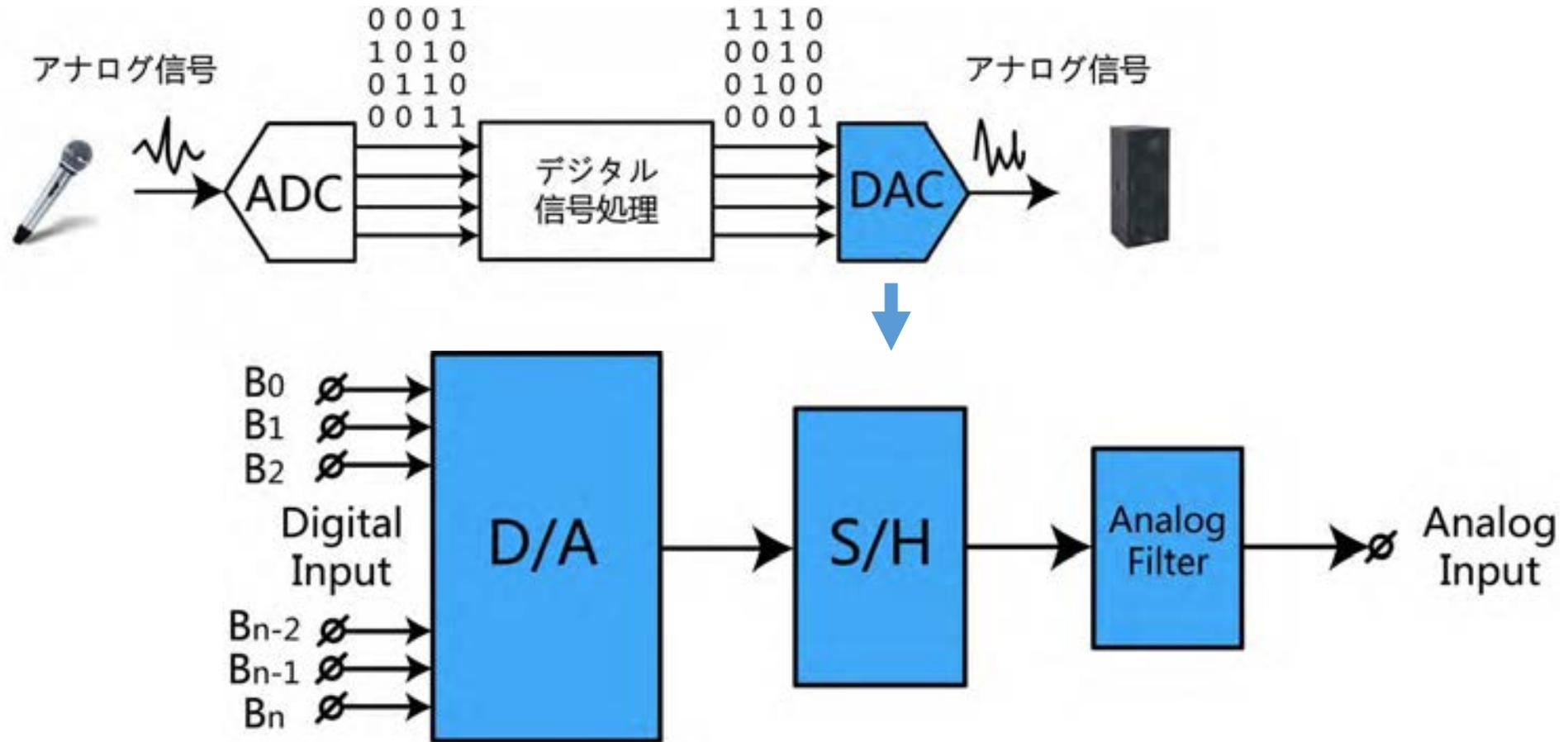
目次

- I. 研究背景・目的
- II. 提案するGray-code入力のDACの構成と動作
- III. SPICEによるシミュレーション検証
- IV. まとめ

目次

- I. 研究背景・目的
- II. 提案するGray-code入力のDACの構成と動作
- III. SPICEによるシミュレーション検証
- IV. まとめ

デジタル-アナログコンバータ(DAC、ダック)



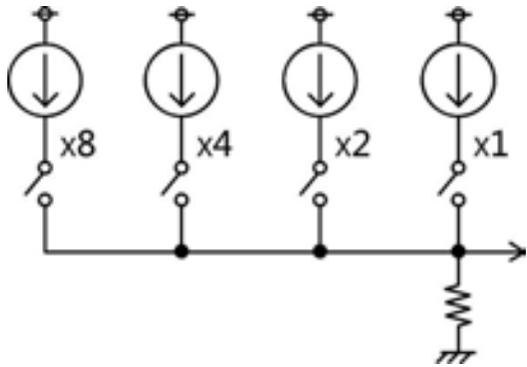
研究目的

Gray-codeを入力としての綺麗な構成のDACは**実現が難しい**と考えられてきた.

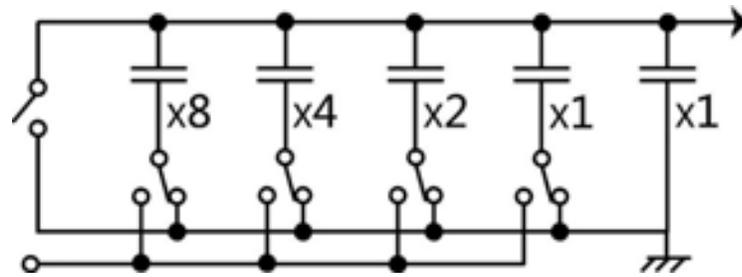


本論文では**Gray-code入力のDAC**が実現できることを示す.

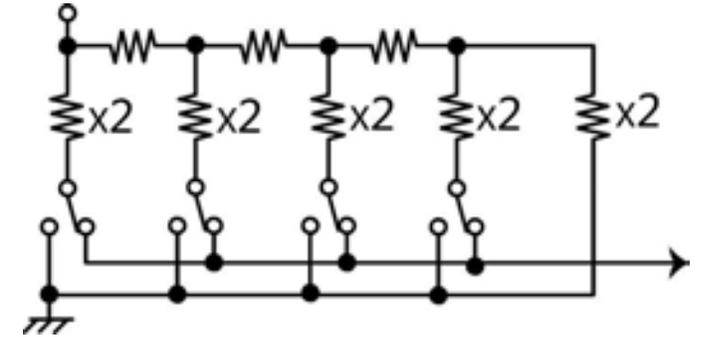
D/A変換器の基本的なアーキテクチャ



電流源型DAC



容量型DAC



抵抗型DAC

スイッチは2進数(バイナリコード、Binary code)で駆動



グリッチ

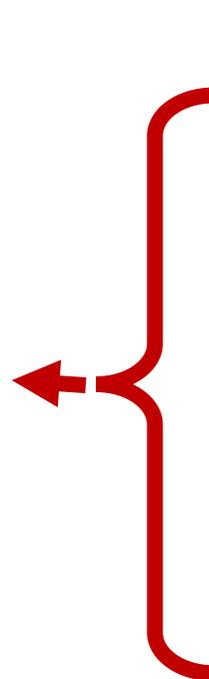
グリッチ (Glitch) の影響



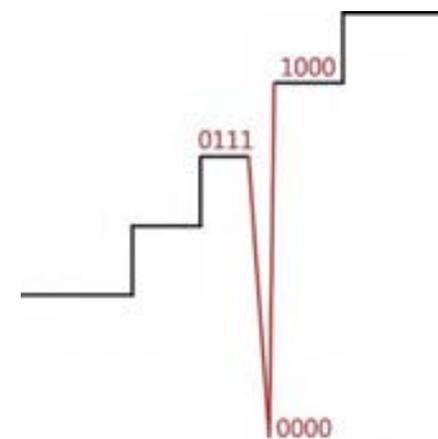
グラフィックディスプレイでのグリッチ

グリッチの発生原理

Decimal numbers	Natural Binary code
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1
10	1 0 1 0
11	1 0 1 1
12	1 1 0 0
13	1 1 0 1
14	1 1 1 0
15	1 1 1 1

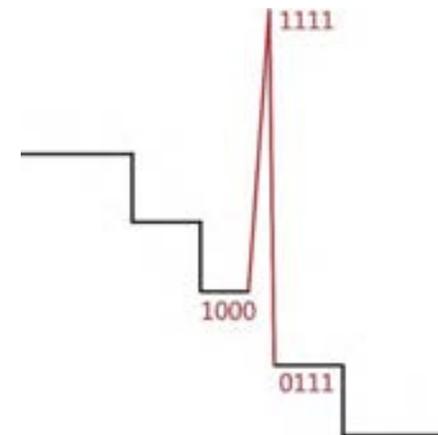


7→8
の時



0111→0110→0100→**0000**→1000

8→7
の時

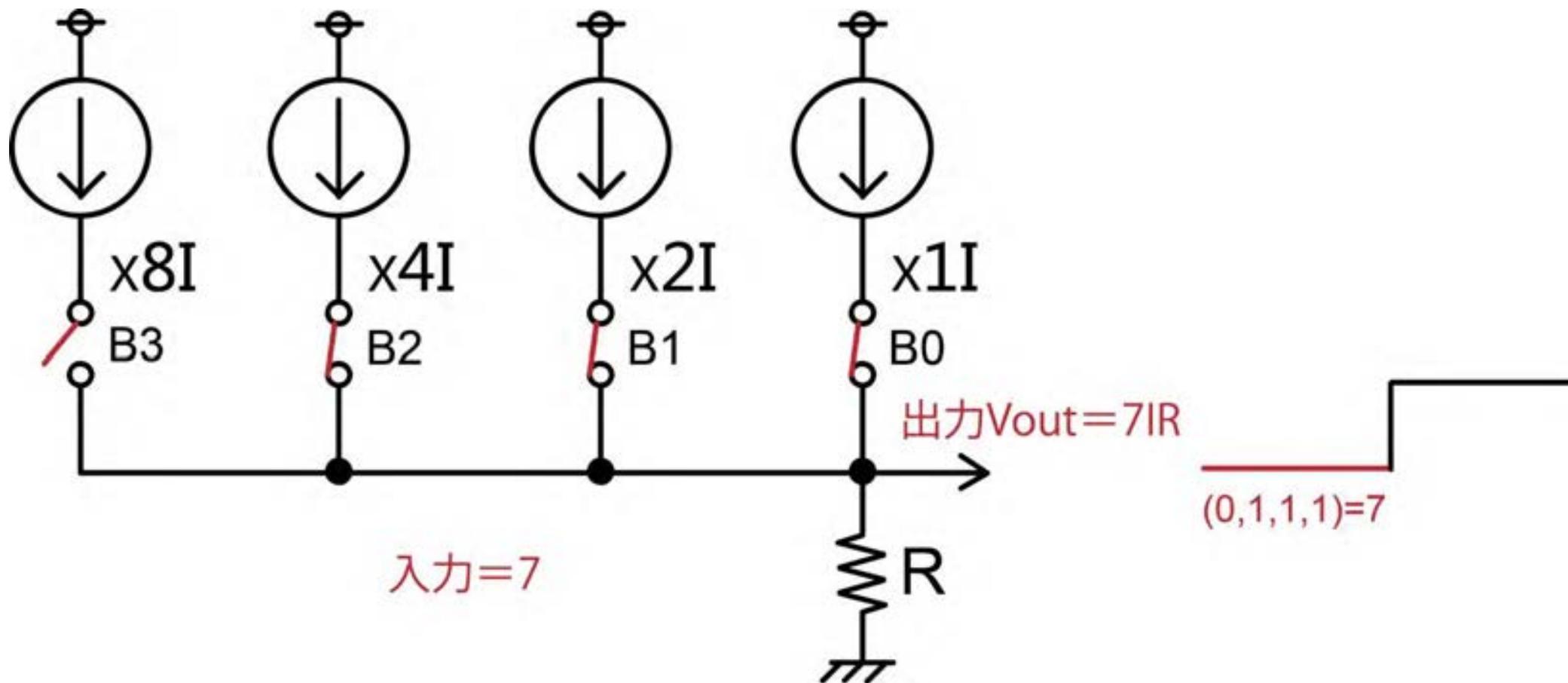


1000→1001→1011→**1111**→0111

最上位ビット (MSB) が変化 (中央値の付近)

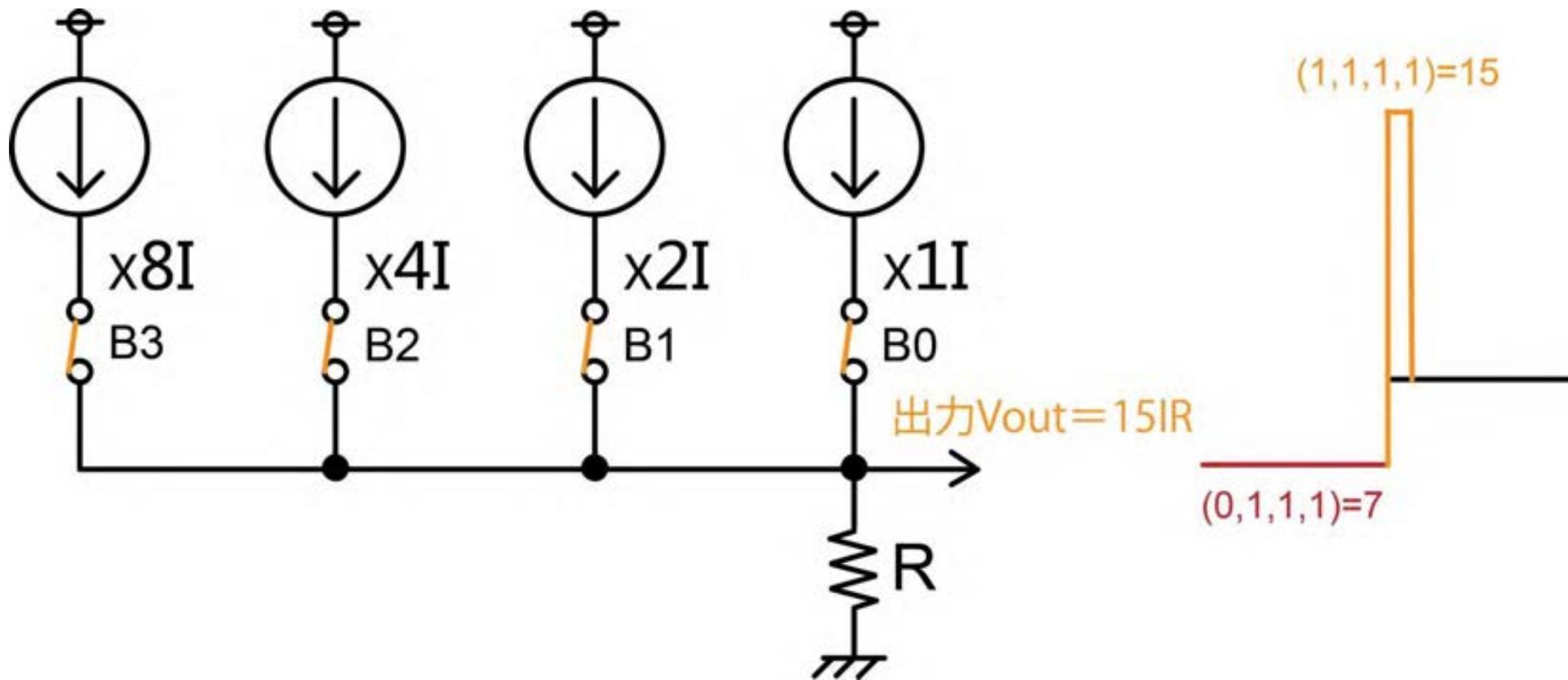
ワーストケース

スイッチ切り替えタイミング



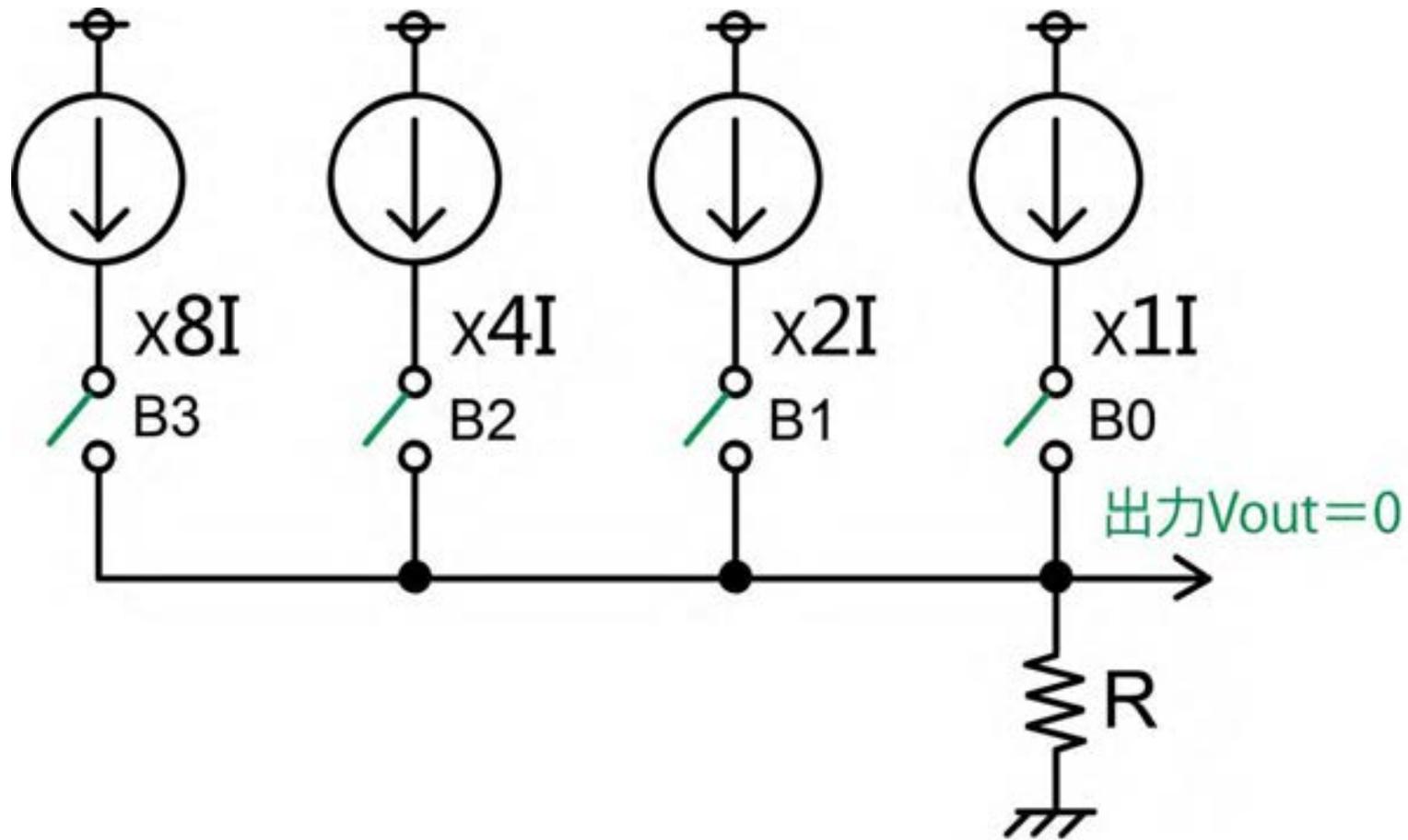
入力は7→8の時

スイッチ切り替えタイミング その1

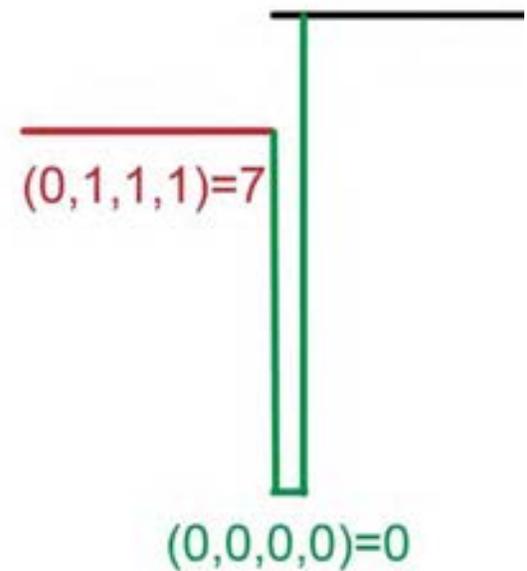


$B3$ が最初にスイッチングすると

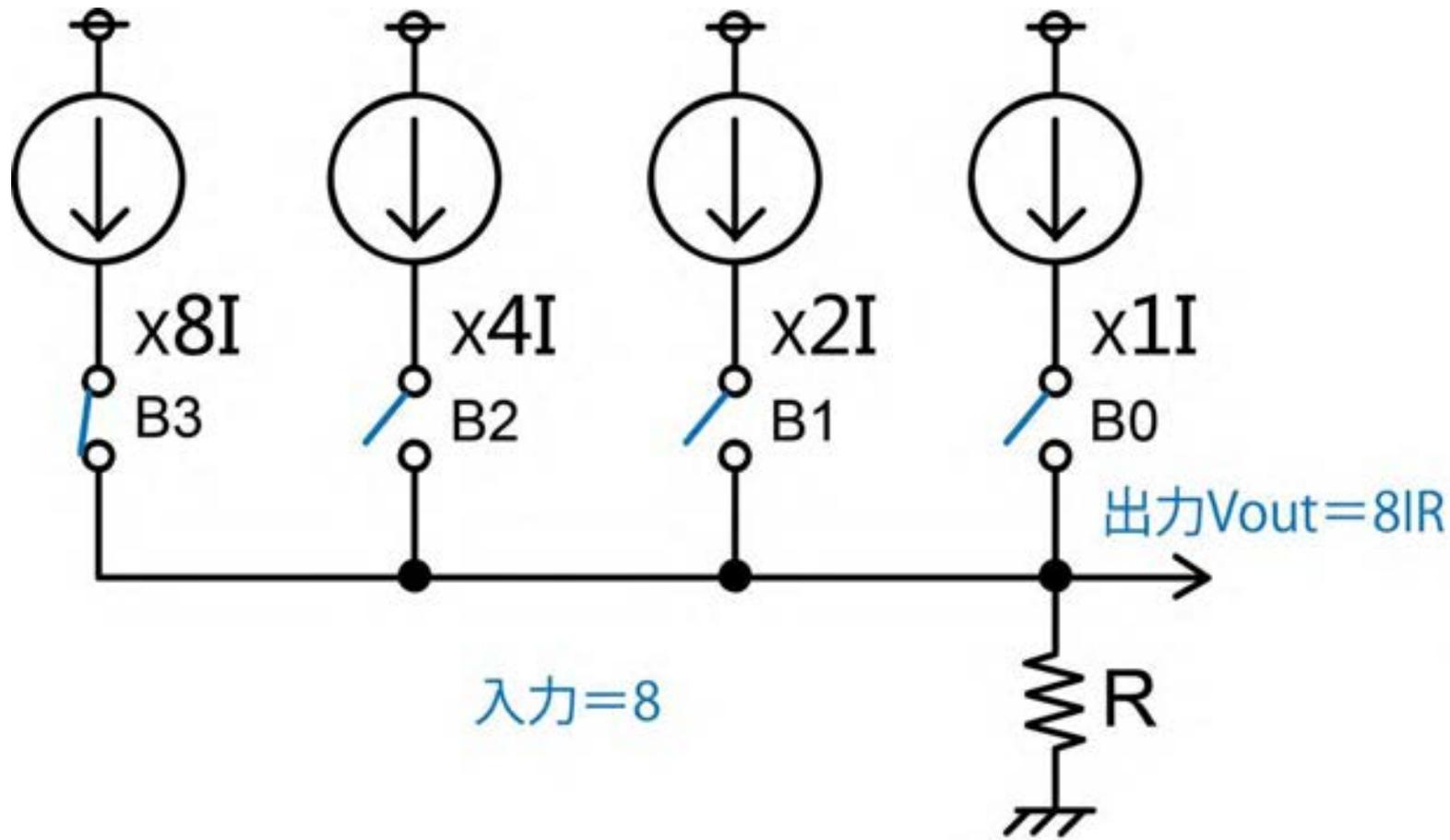
スイッチ切り替えタイミング その1



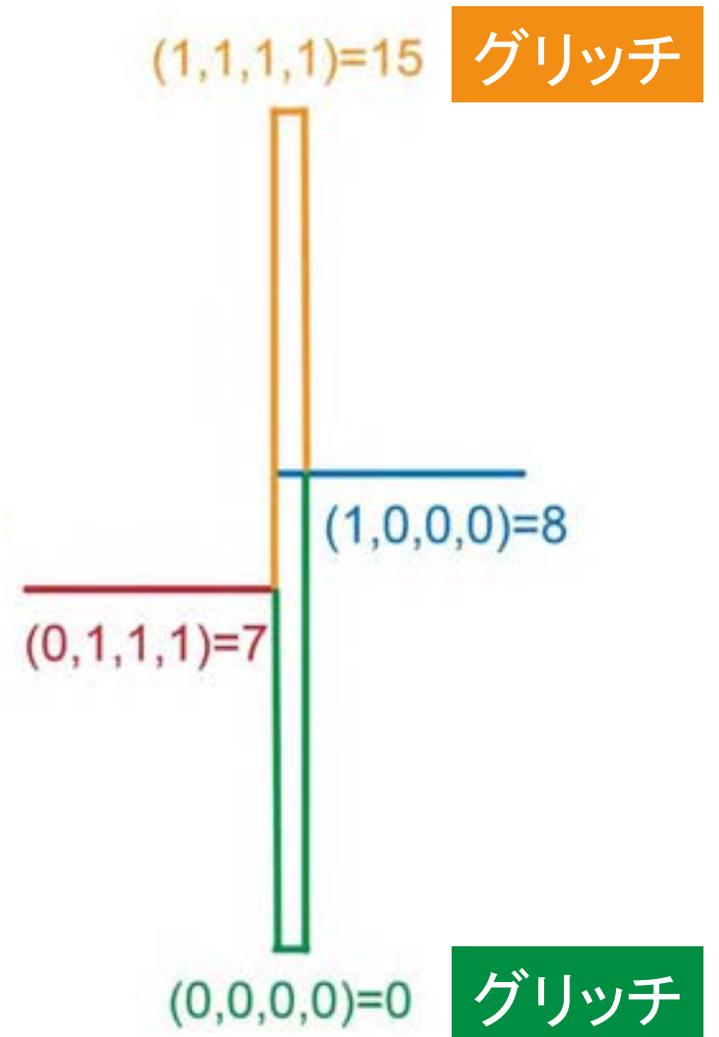
B3が最後にスイッチングすると



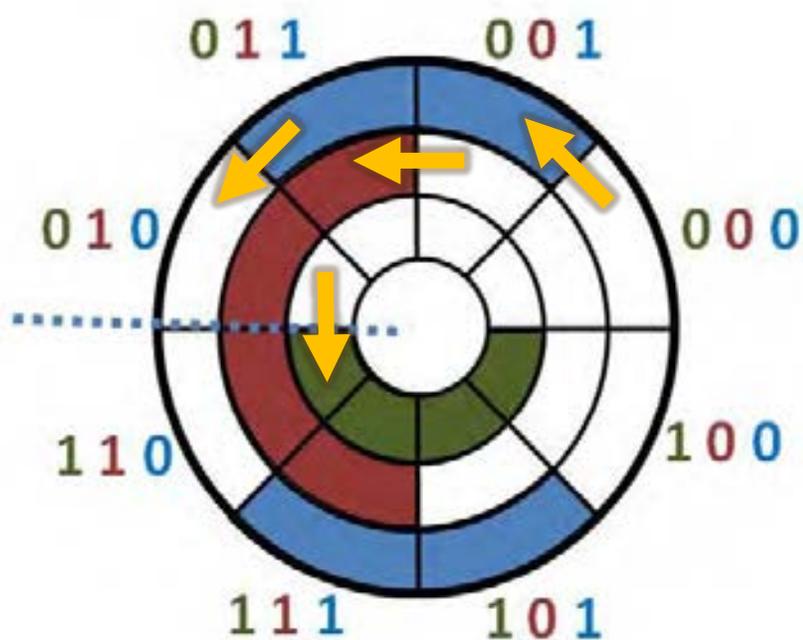
スイッチングスキュー → グリッチ



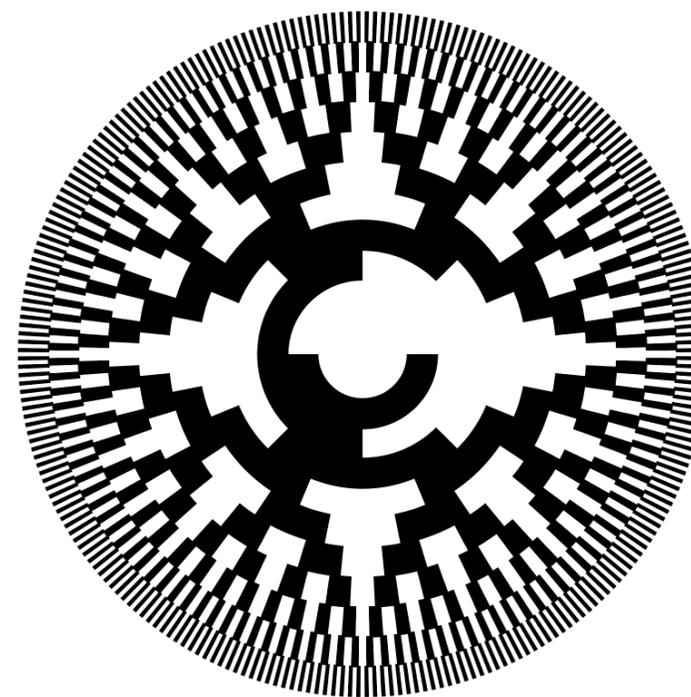
入力は7→8の時



グレイコード (Gray-code)



$Q_2 Q_1 Q_0$
000
001
011
010
110
111
101
100



グレイコード: 前後に隣接する符号間のハミング距離が必ず1

ベル研究所のフランク・グレイが1947年の特許出願書で最初に使用した。

グレイコードD/A変換器

Binary code と Gray code の変換が容易 (EXOR)
($G_n = B_{n+1} \oplus B_n$)

Gray code 入力のDA変換器



グリッチが小さくできる

Binary code: 0111 → 0110 → 0100 → 0000 → 1000

Gray code: 0100 → 1100

Decimal numbers	Natural Binary code	Gray code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

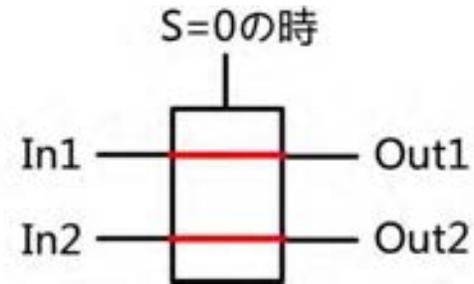
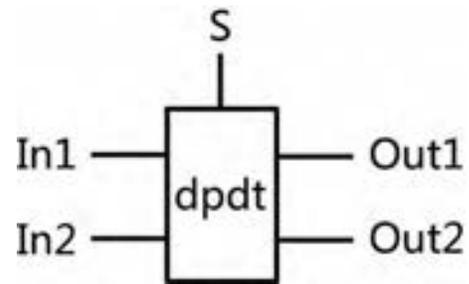
目次

- I. 研究背景・目的
- II. 提案するGray-code入力のDACの構成と動作**
- III. SPICEによるシミュレーション検証
- IV. まとめ

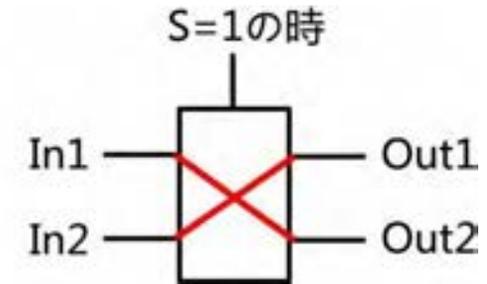
提案するGray-code入力のDACの構成と動作

1. Gray-code入力の電流源型DAC (I-DAC)
2. Gray-code入力の容量型DAC (C-DAC)
3. Gray-code入力の電圧加算型DAC (V-DAC)

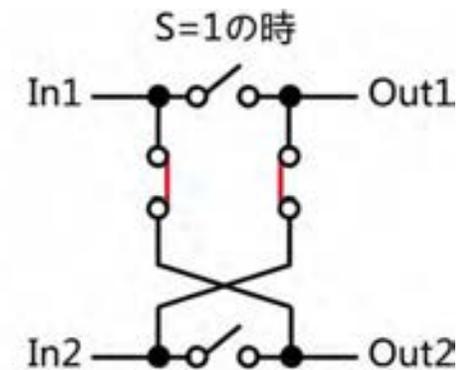
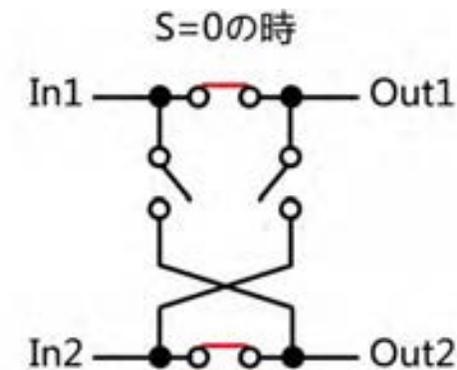
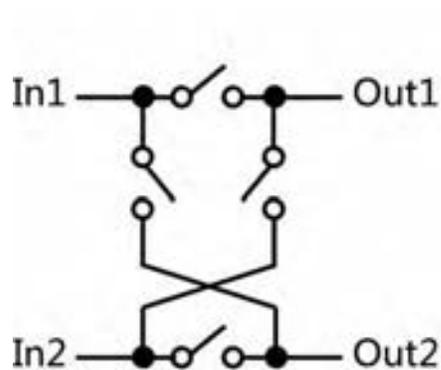
電流/電圧スイッチマトリックス



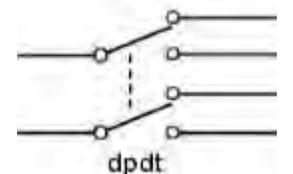
パラレル接続



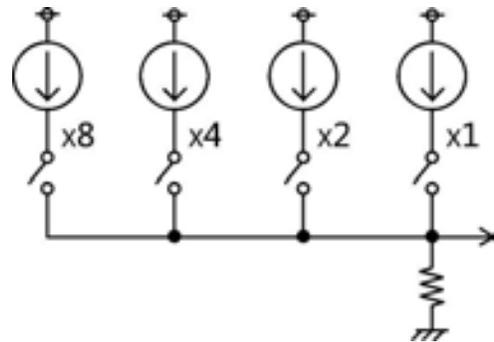
クロス接続



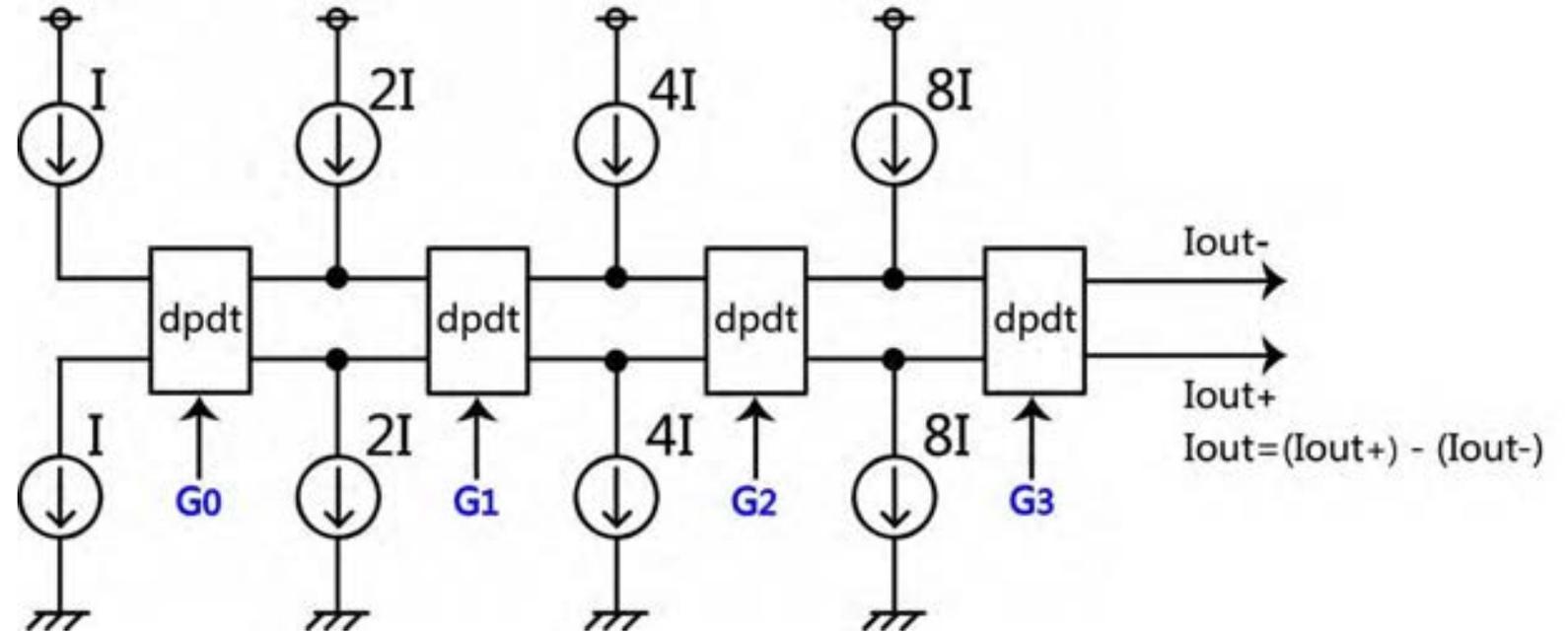
double-pole double-throw (dpdt, 双極双投) スイッチで実現



1. Gray-code入力の電流源型DAC

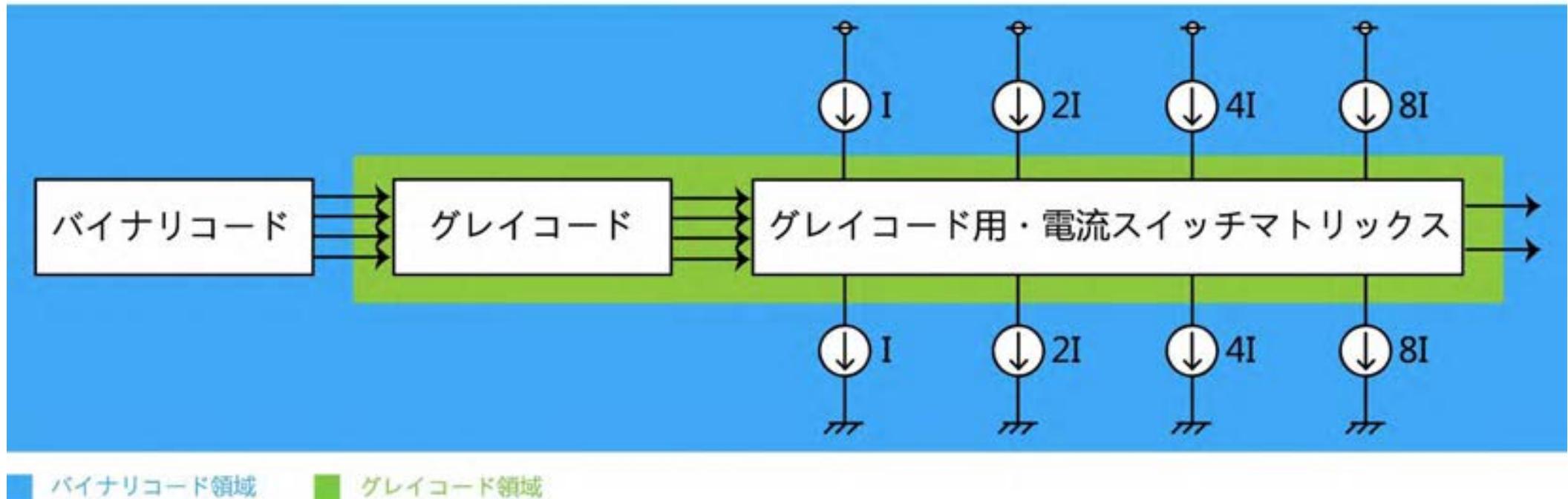


従来の電流源型DAC



Gray-code入力の電流源型DAC

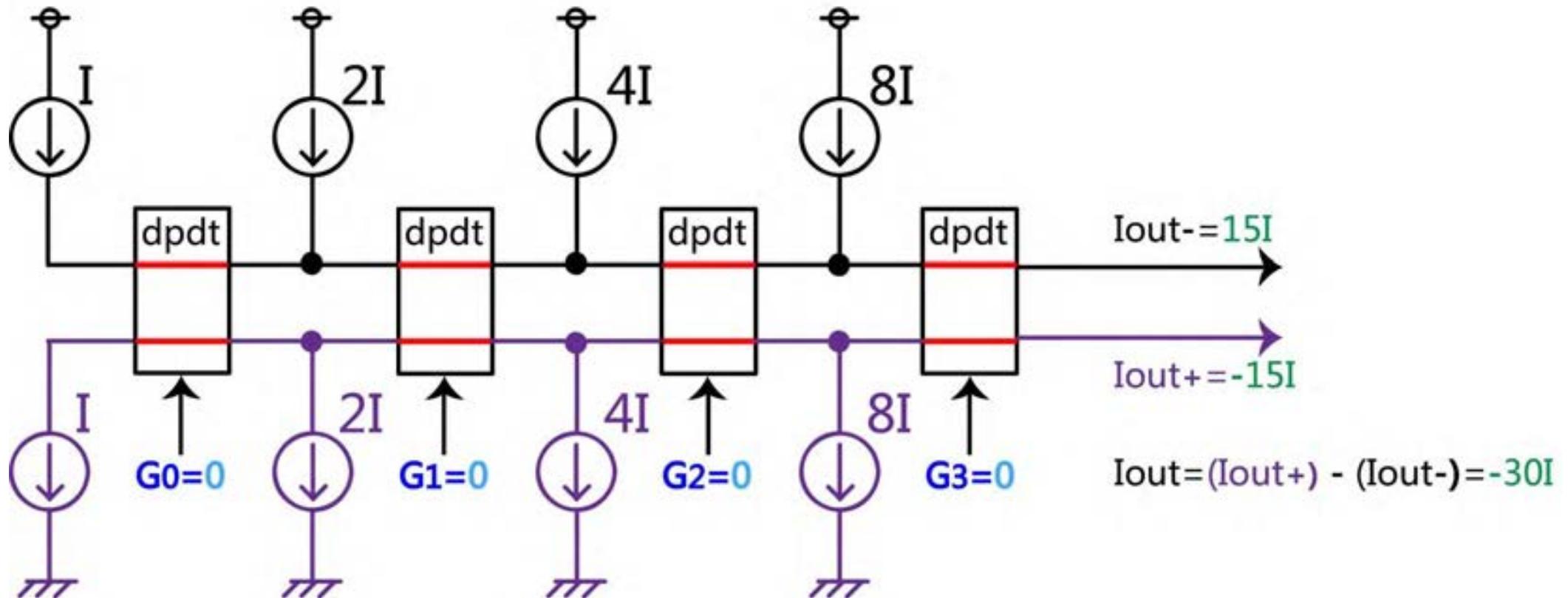
コード変換



バイナリの電流源順番に対応できるGray-code入力

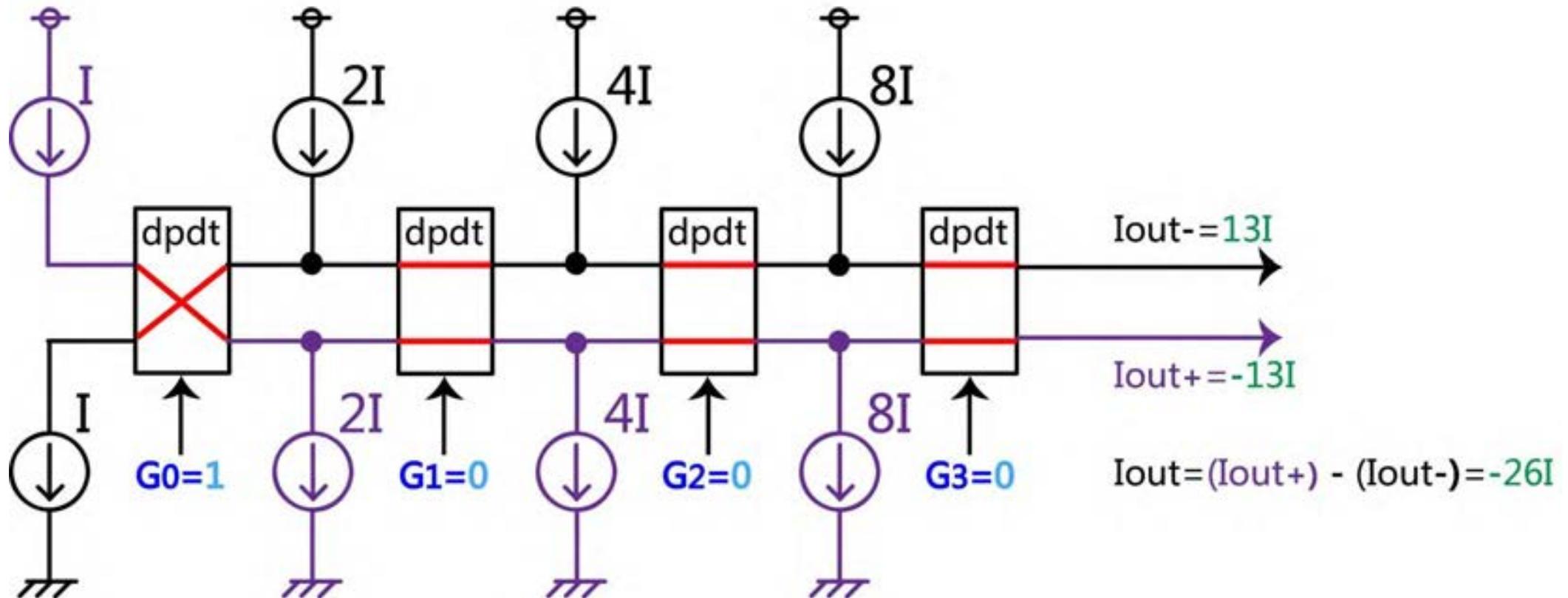
Gray-code入力のI-DACの動作 (データ=0の場合)

eg. Data=0



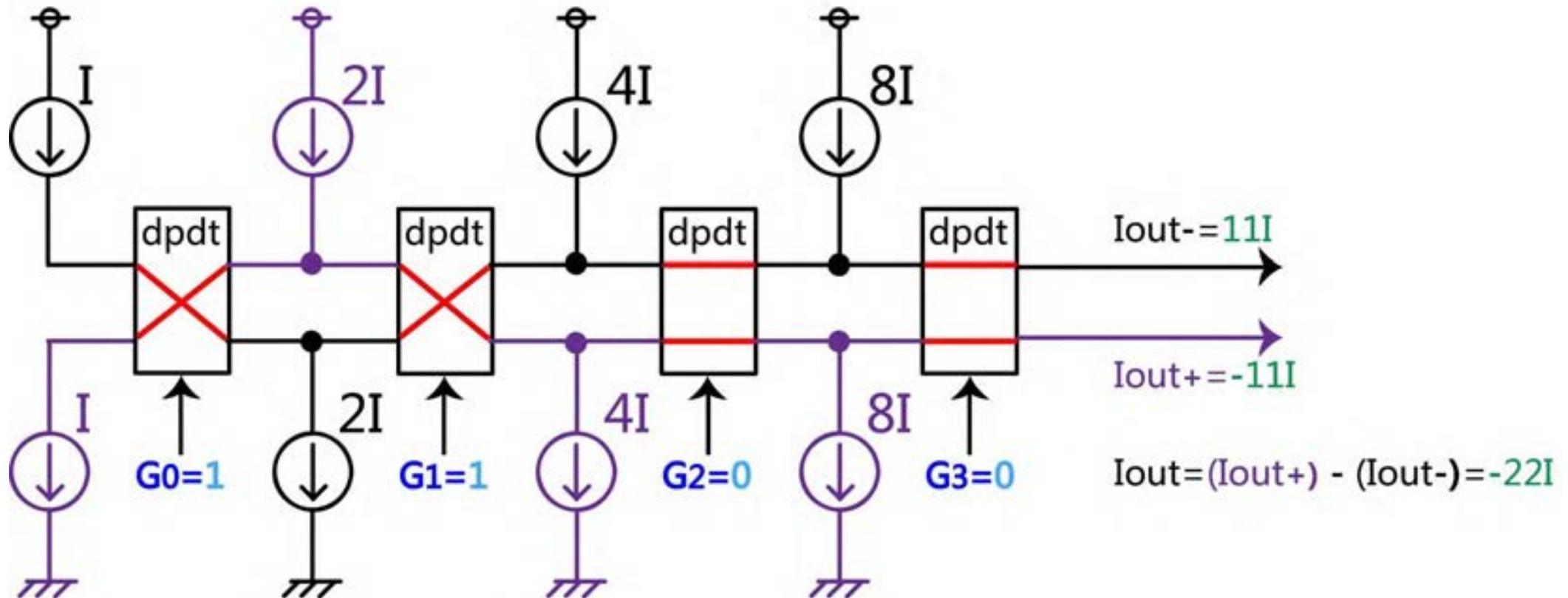
Gray-code入力のI-DACの動作 (データ=1の場合)

eg. Data=1



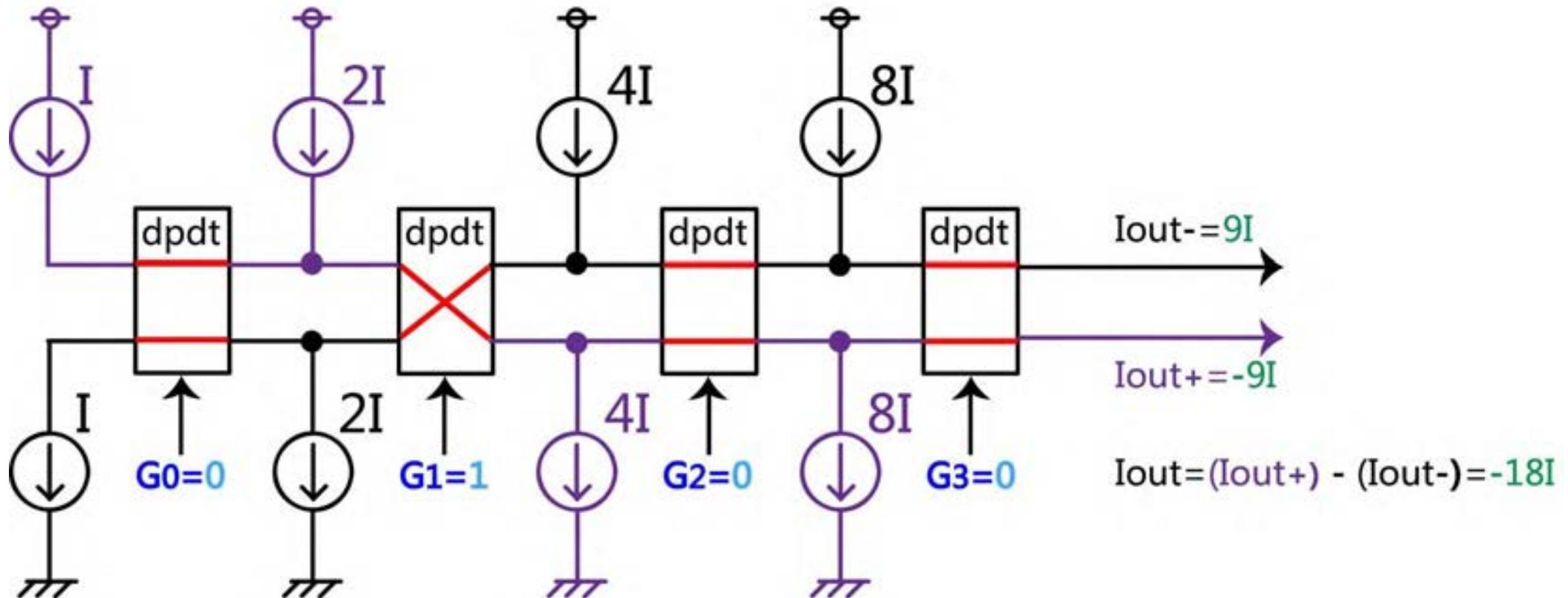
Gray-code入力のI-DACの動作 (データ=2の場合)

eg. Data=2



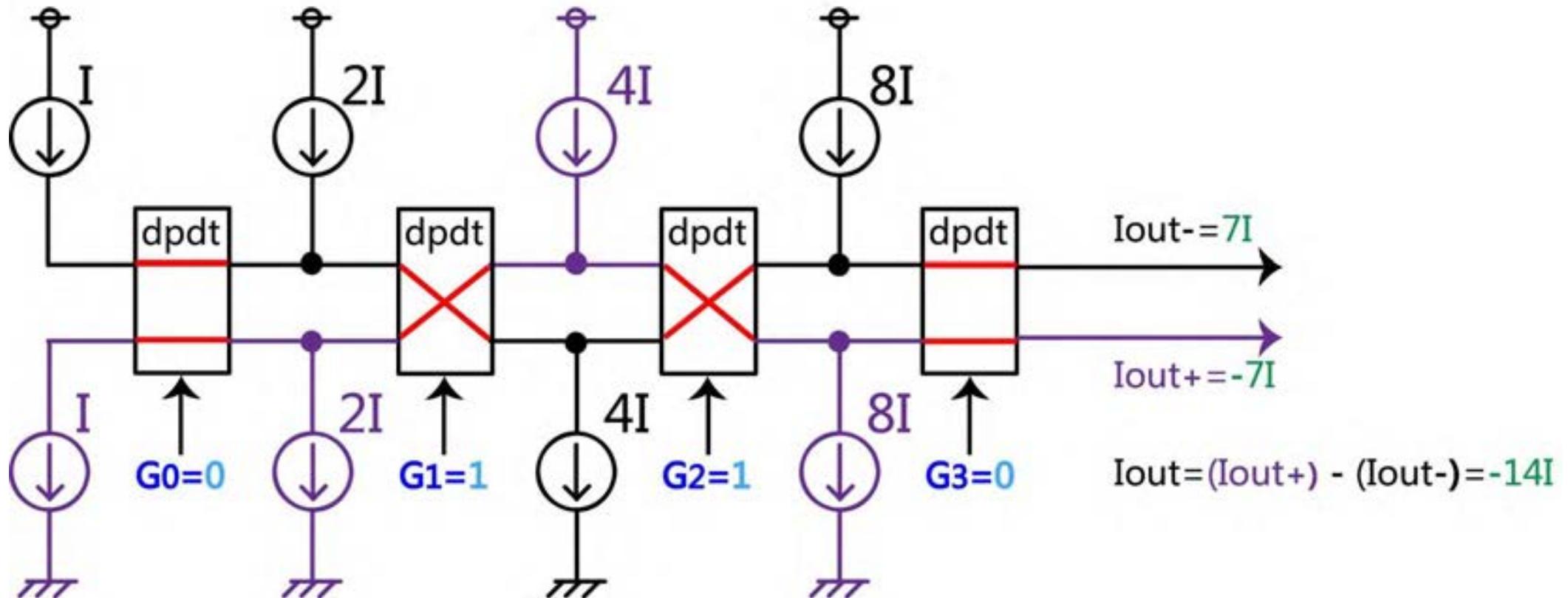
Gray-code入力のI-DACの動作 (データ=3の場合)

eg. Data=3



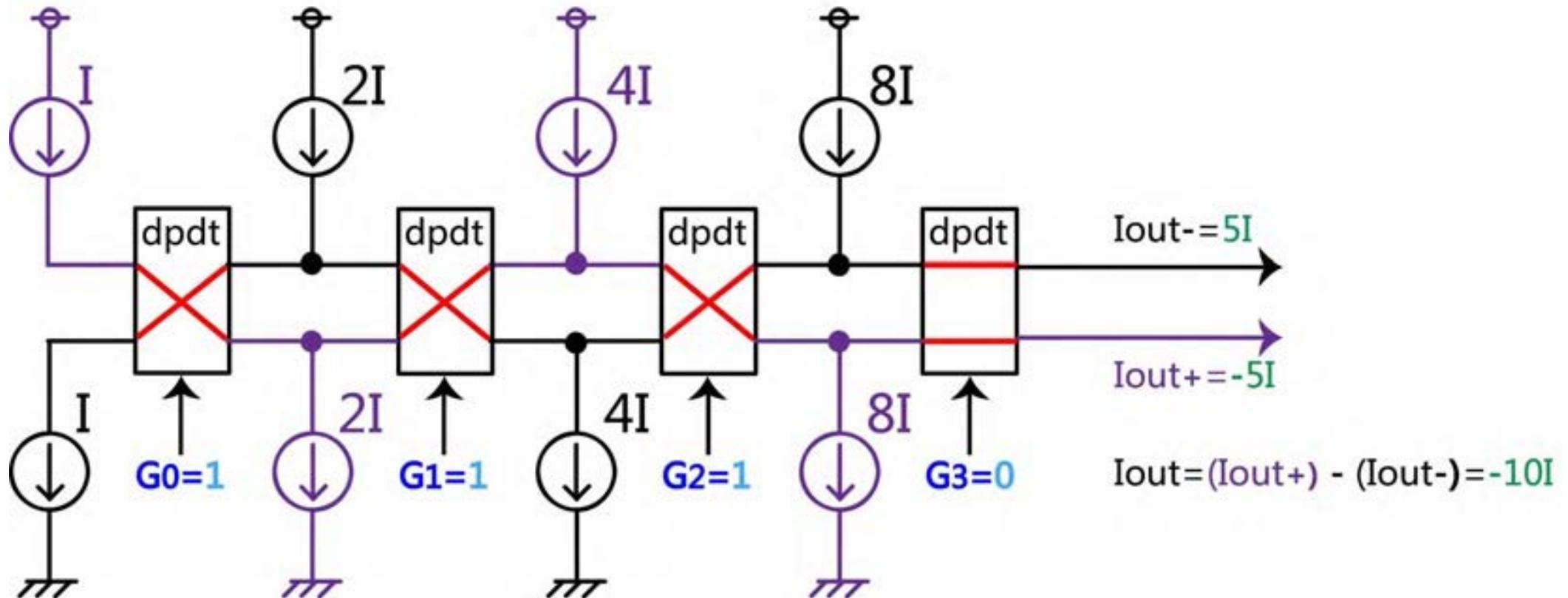
Gray-code入力のI-DACの動作 (データ=4 の場合)

eg. Data=4



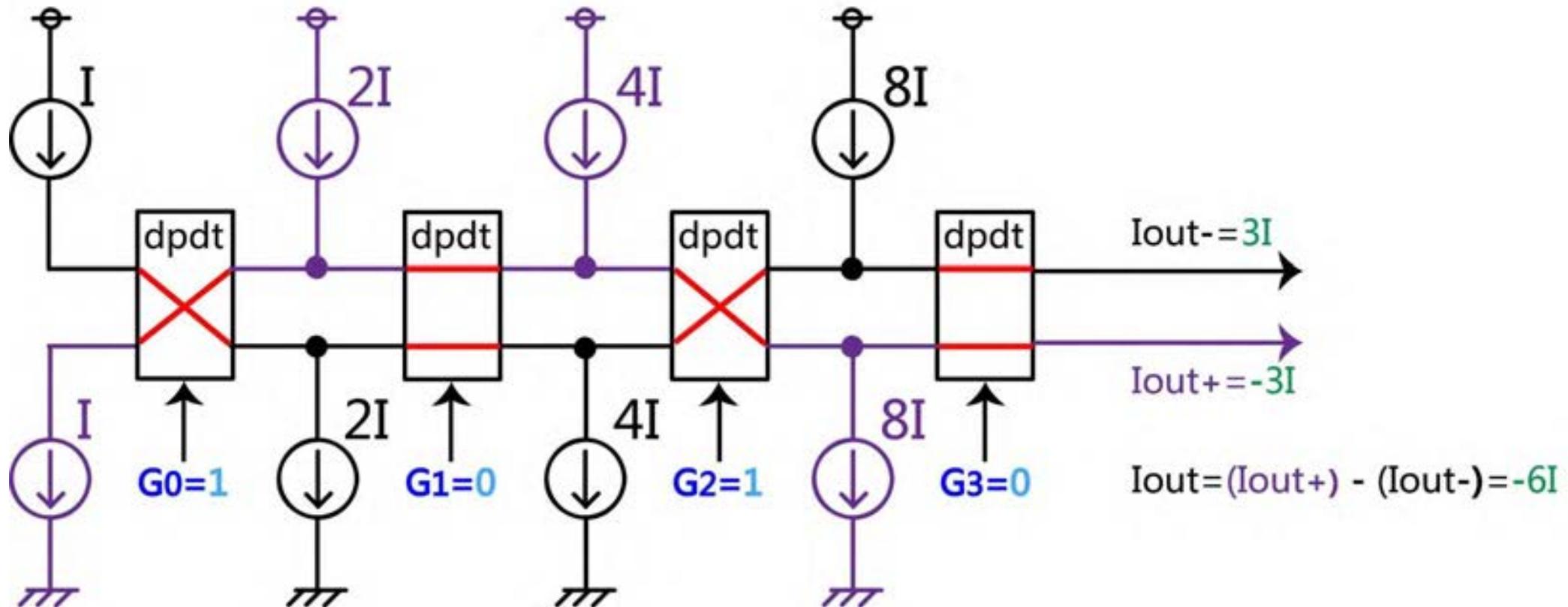
Gray-code入力のI-DACの動作 (データ=5の場合)

eg. Data=5



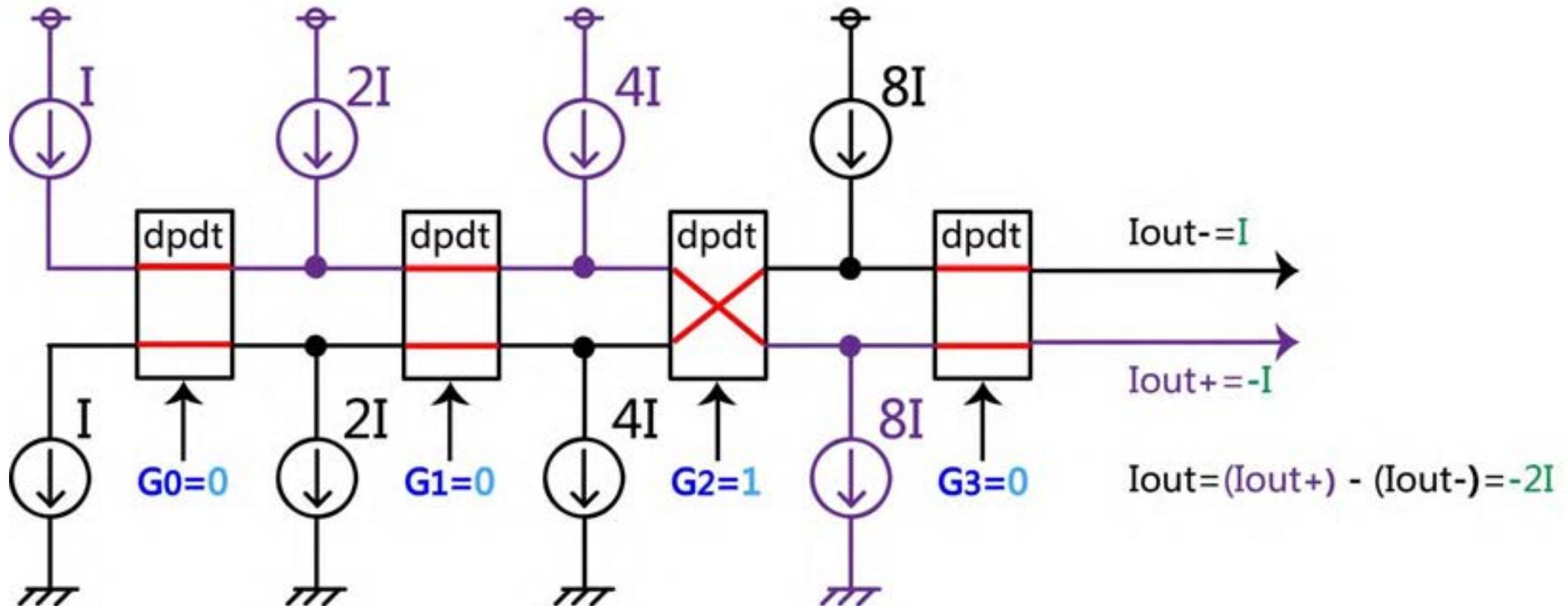
Gray-code入力のI-DACの動作 (データ=6の場合)

eg. Data=6



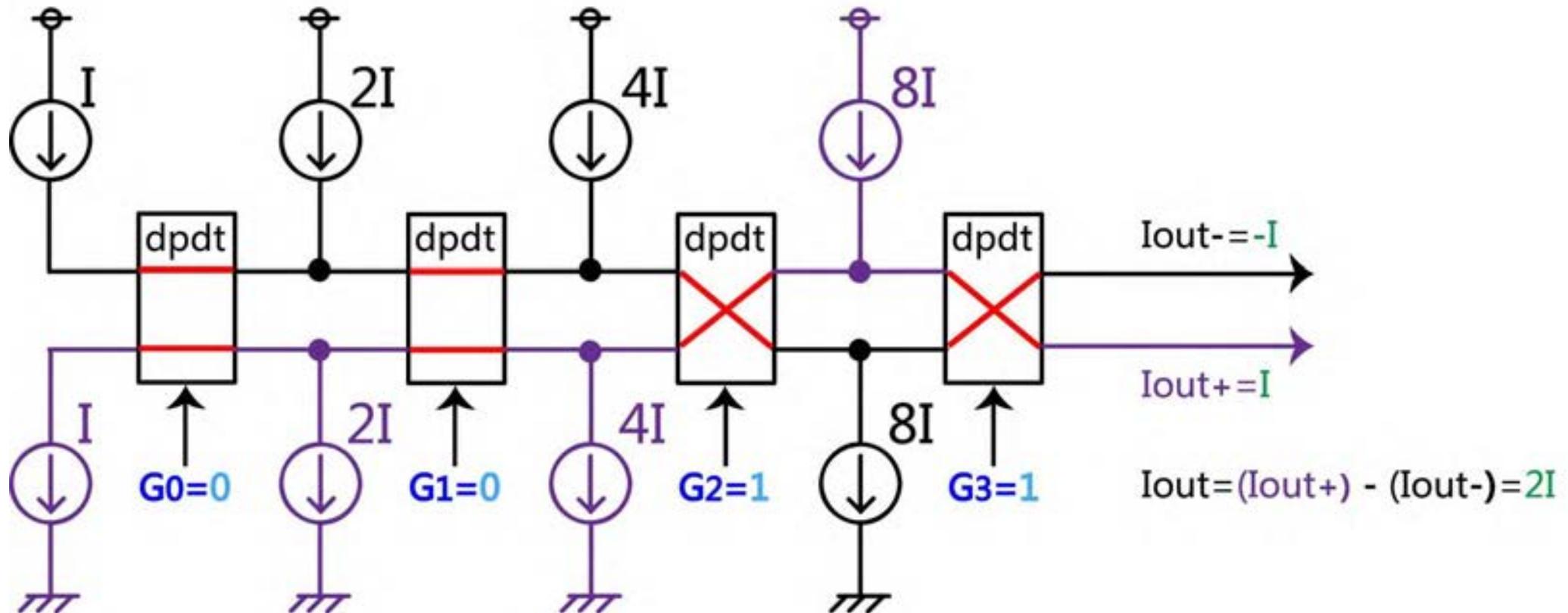
Gray-code入力のI-DACの動作 (データ=7の場合)

eg. Data=7



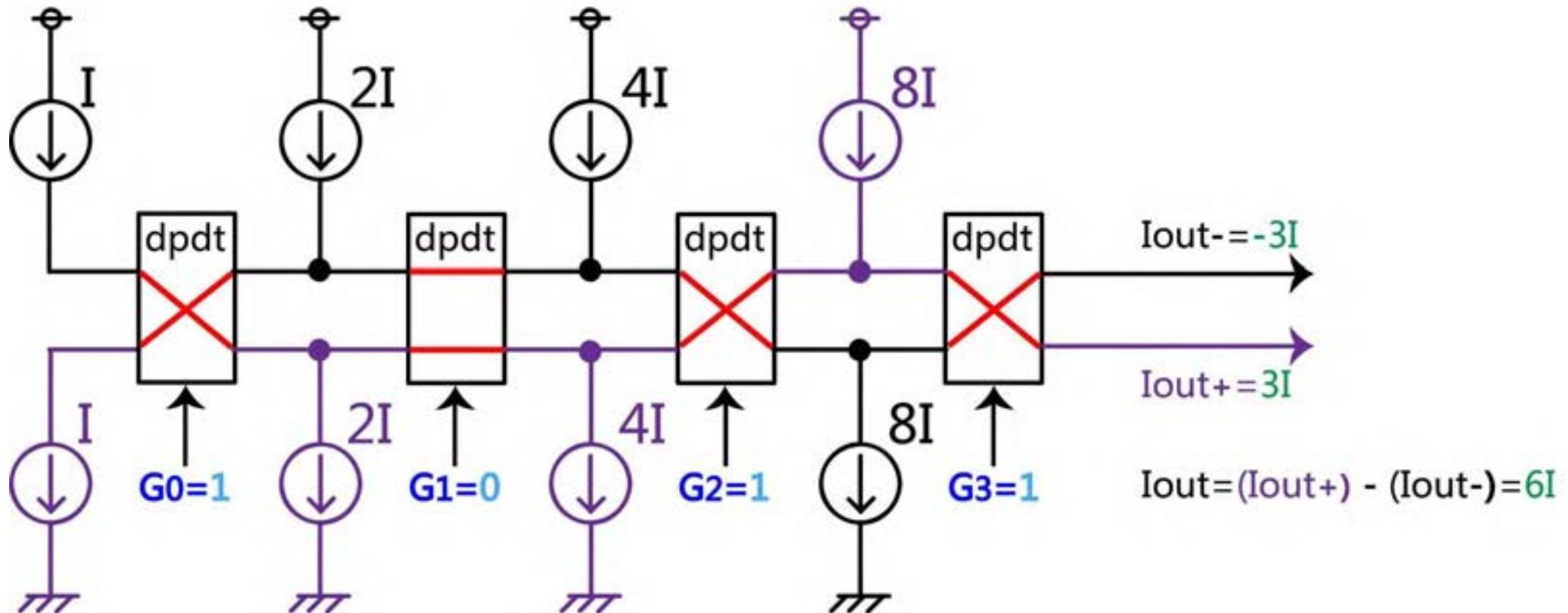
Gray-code入力のI-DACの動作 (データ=8の場合)

eg. Data=8



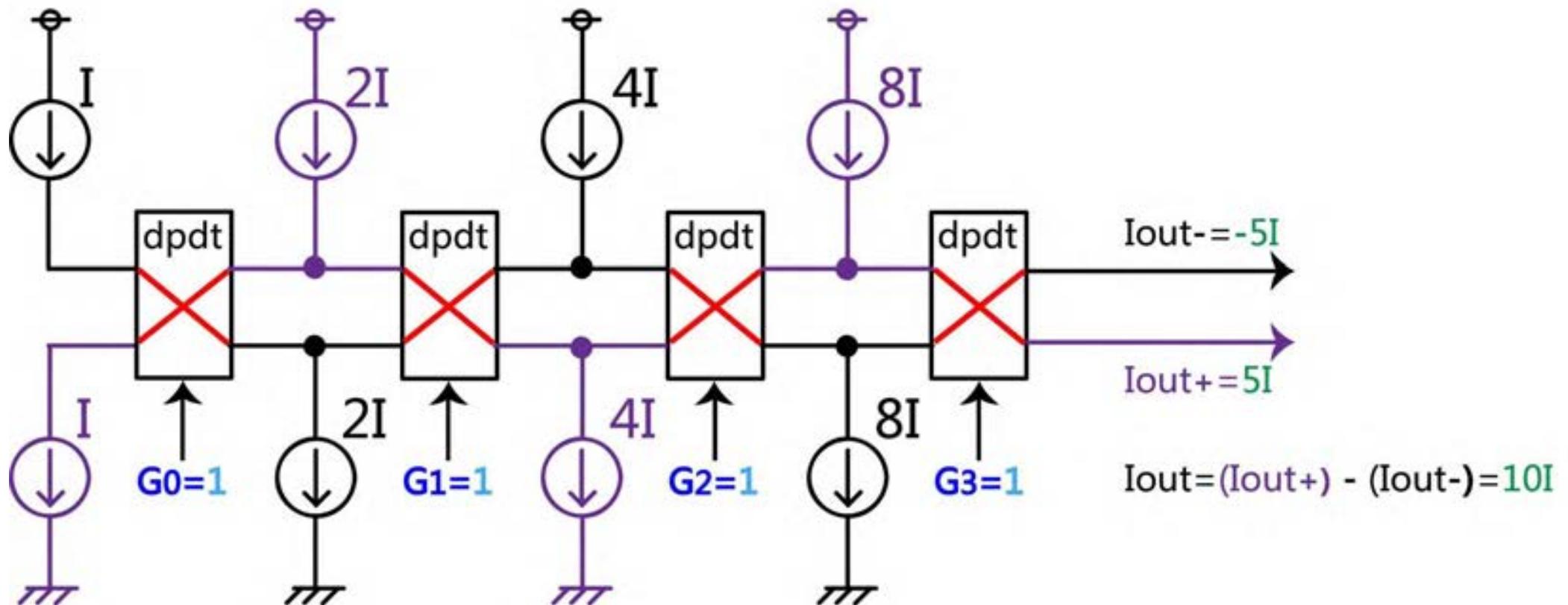
Gray-code入力のI-DACの動作 (データ=9の場合)

eg. Data=9



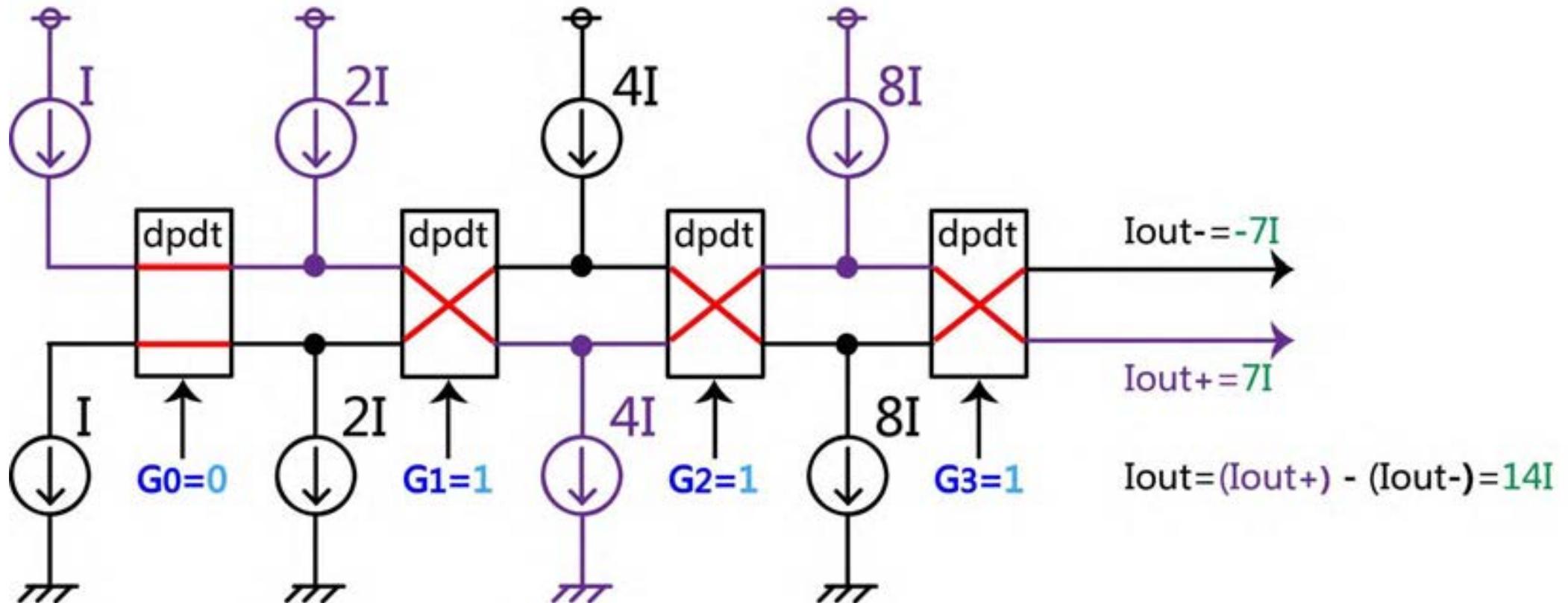
Gray-code入力のI-DACの動作 (データ=10 の場合)

eg. Data=10



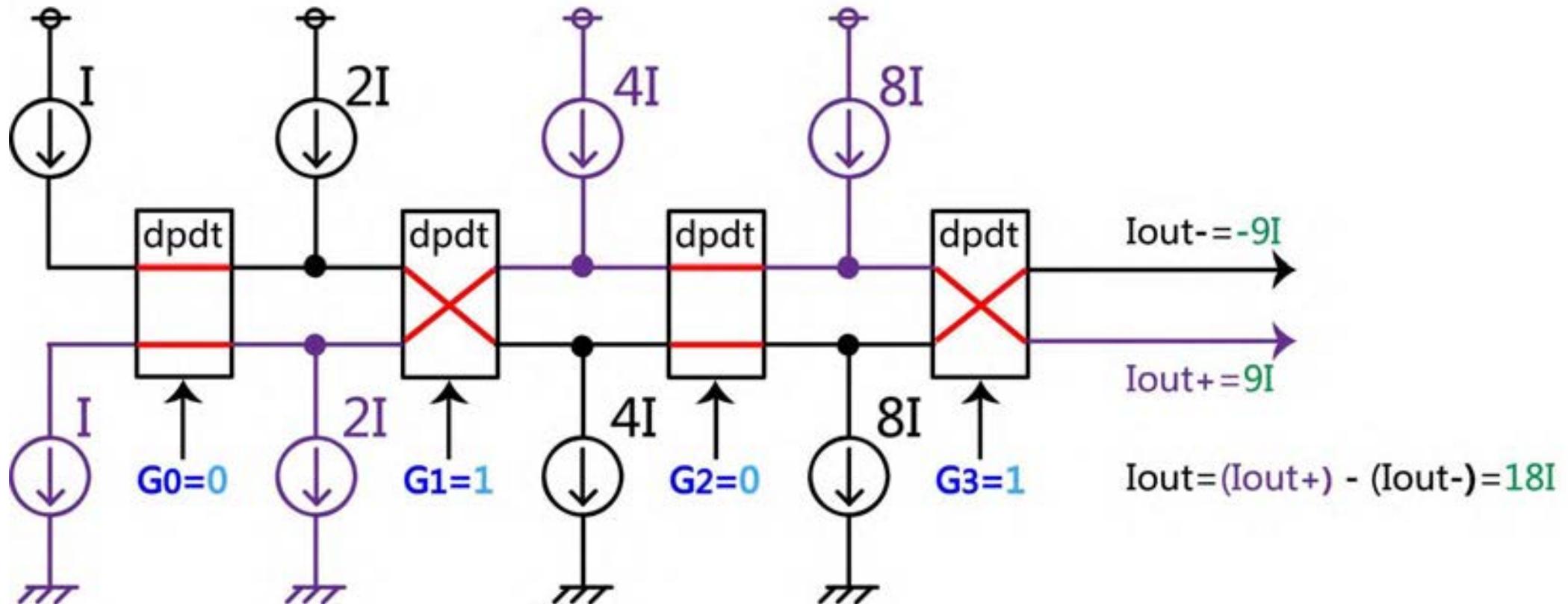
Gray-code入力のI-DACの動作 (データ=11 の場合)

eg. Data=11



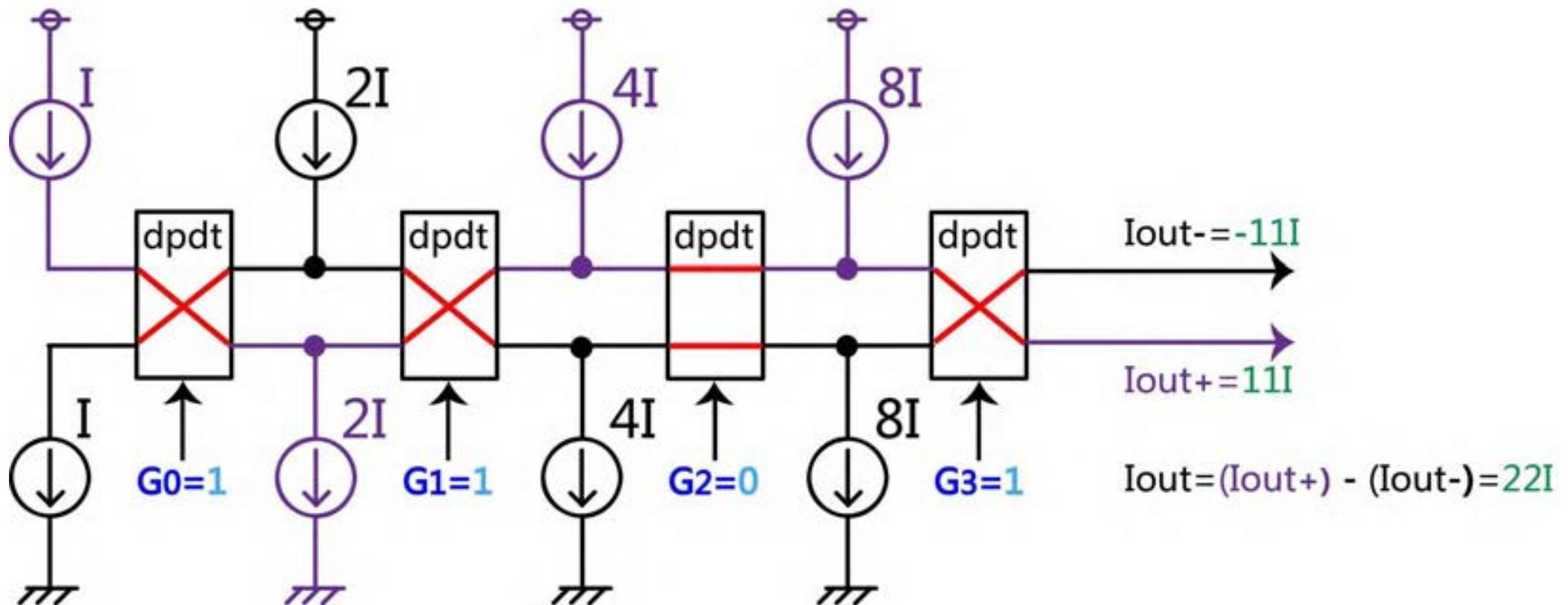
Gray-code入力のI-DACの動作 (データ=12の場合)

eg. Data=12



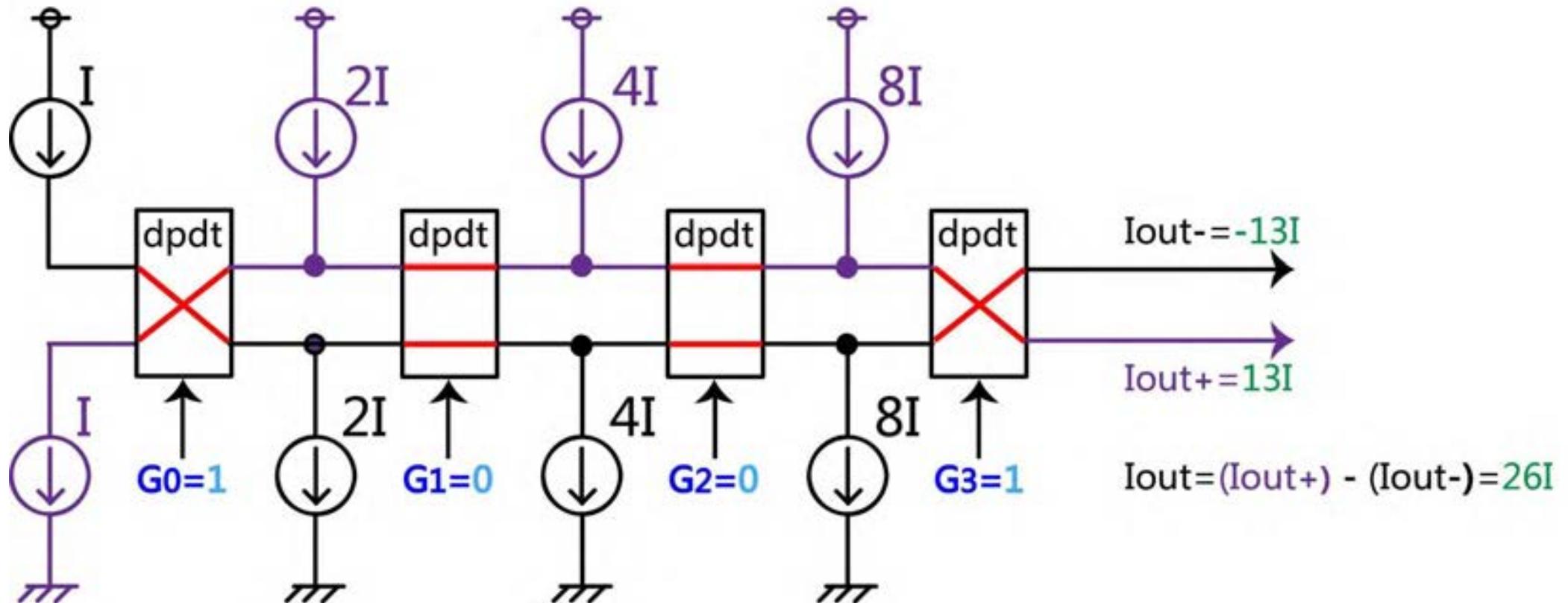
Gray-code入力のI-DACの動作 (データ=13の場合)

eg. Data=13



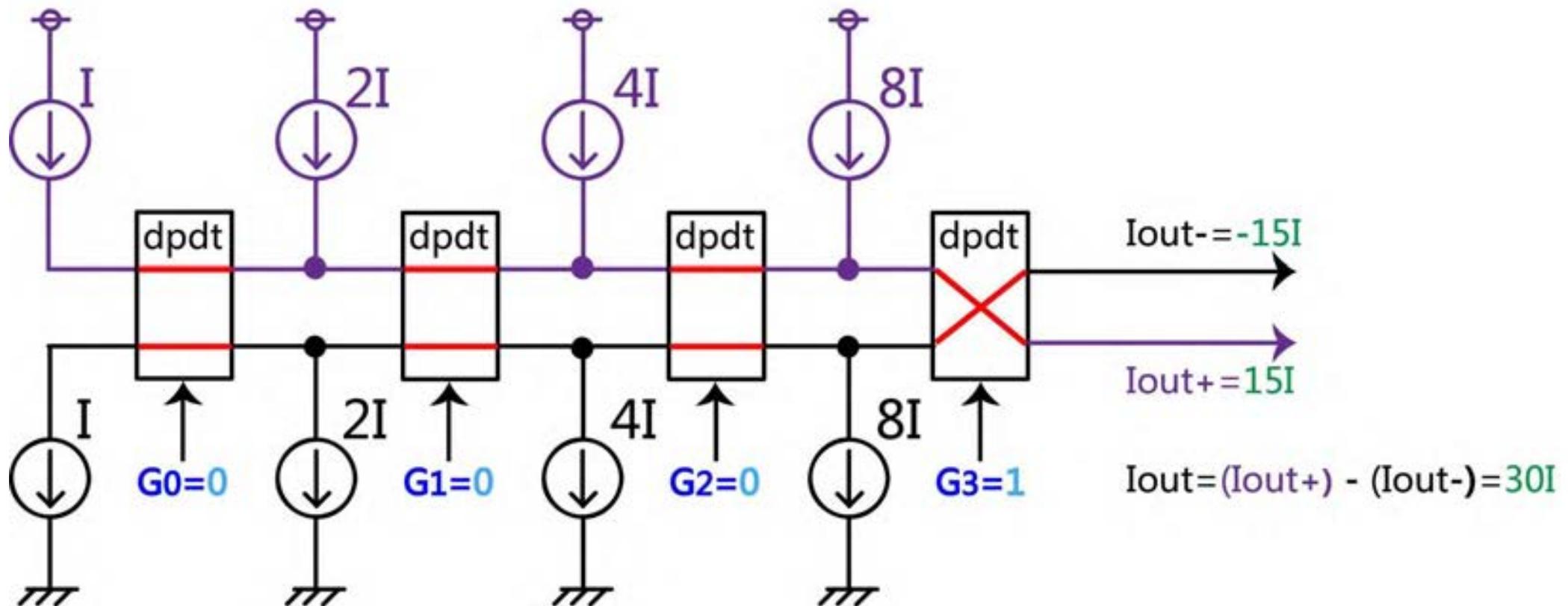
Gray-code入力のI-DACの動作 (データ=14 の場合)

eg. Data=14

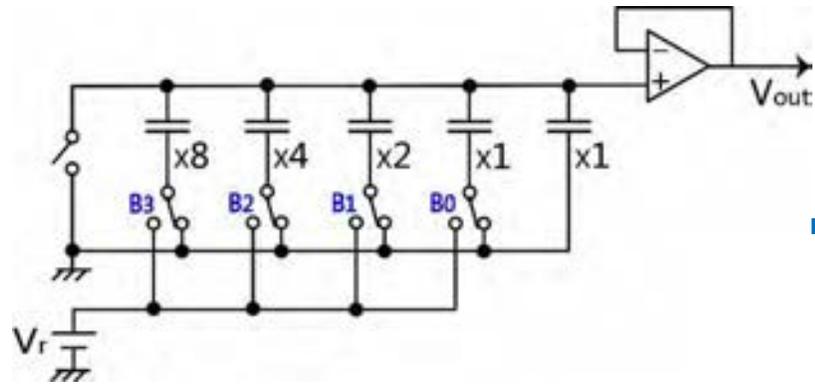


Gray-code入力のI-DACの動作 (データ=15 の場合)

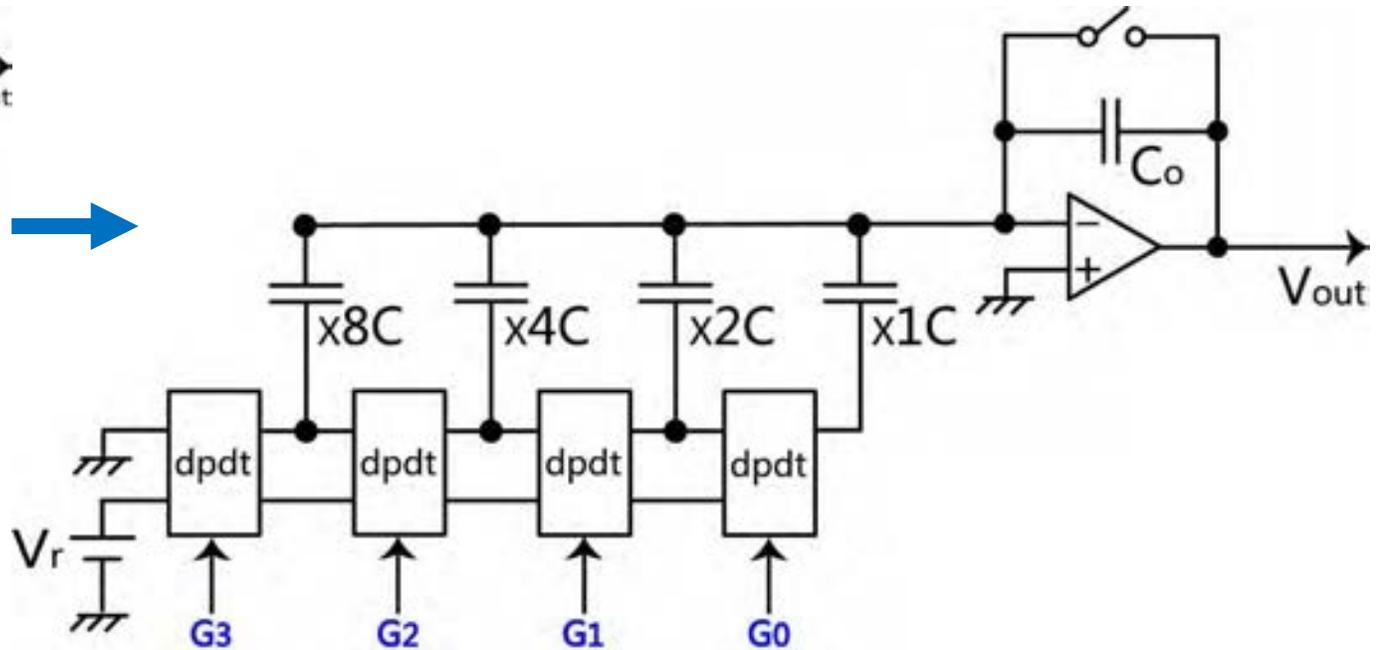
eg. Data=15



2. Gray-code入力の容量型DAC



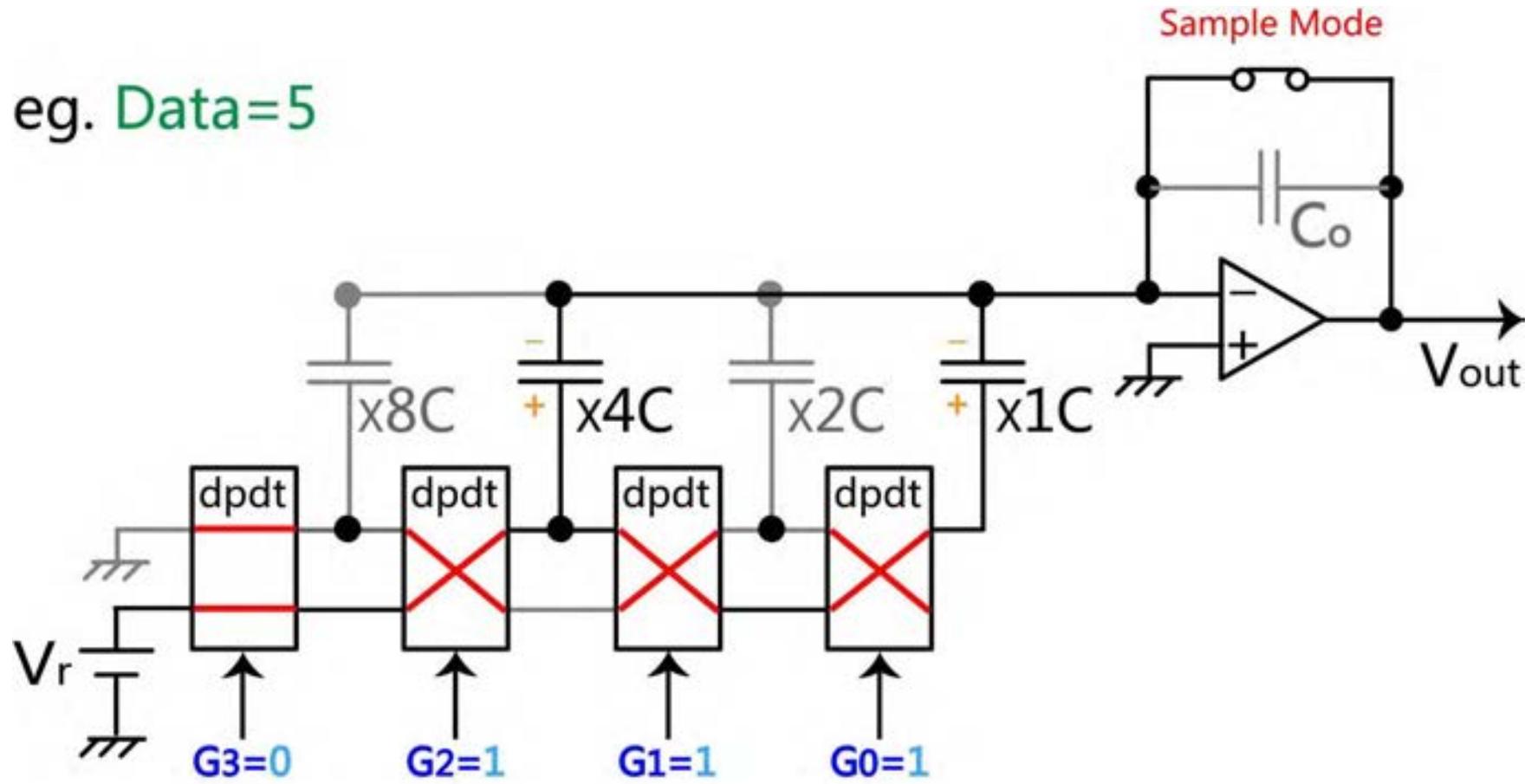
従来の容量型DAC



Gray-code入力の容量型DAC

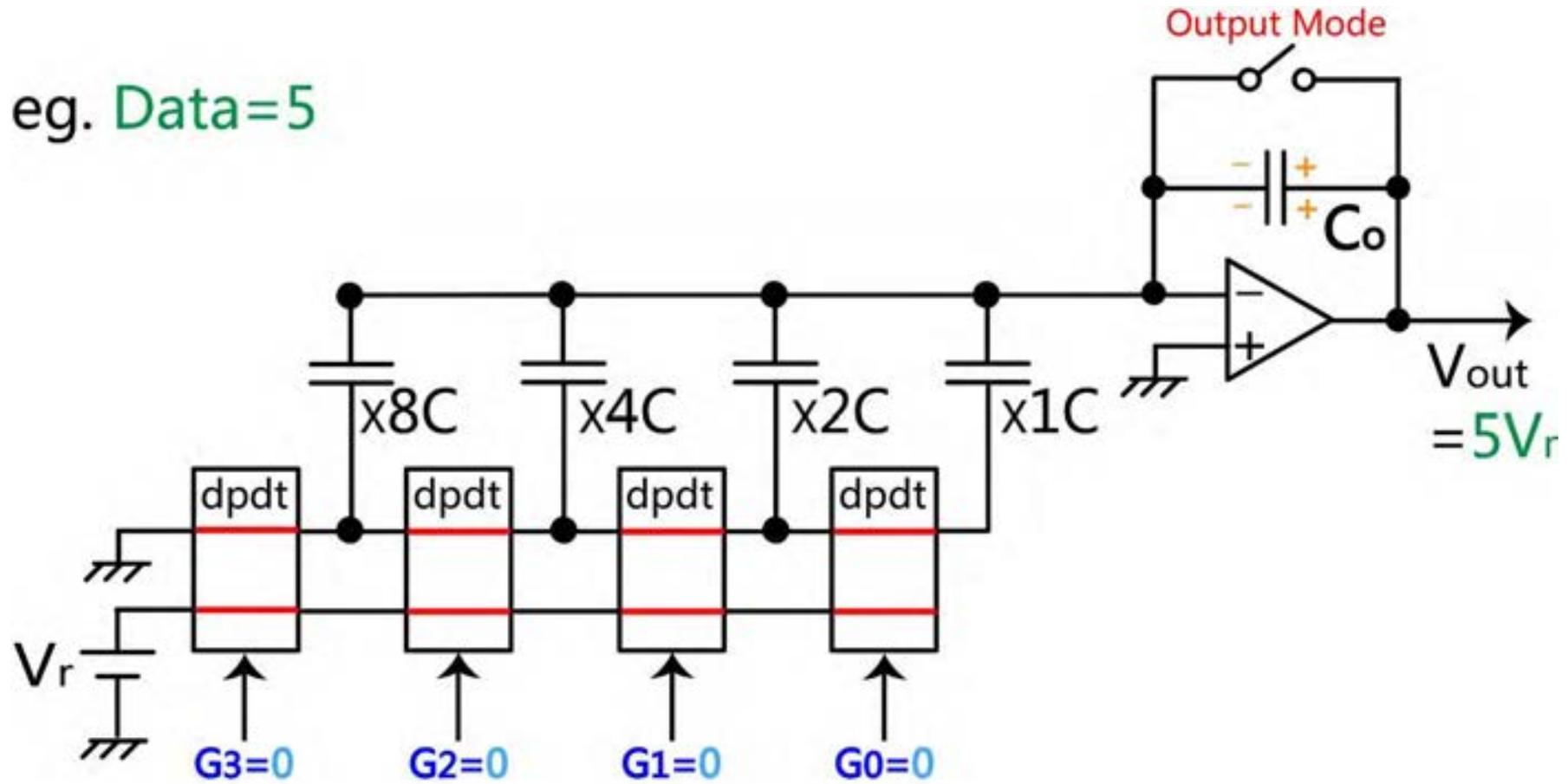
Gray-code入力のC-DACの動作 (データ=5の場合)

eg. Data=5

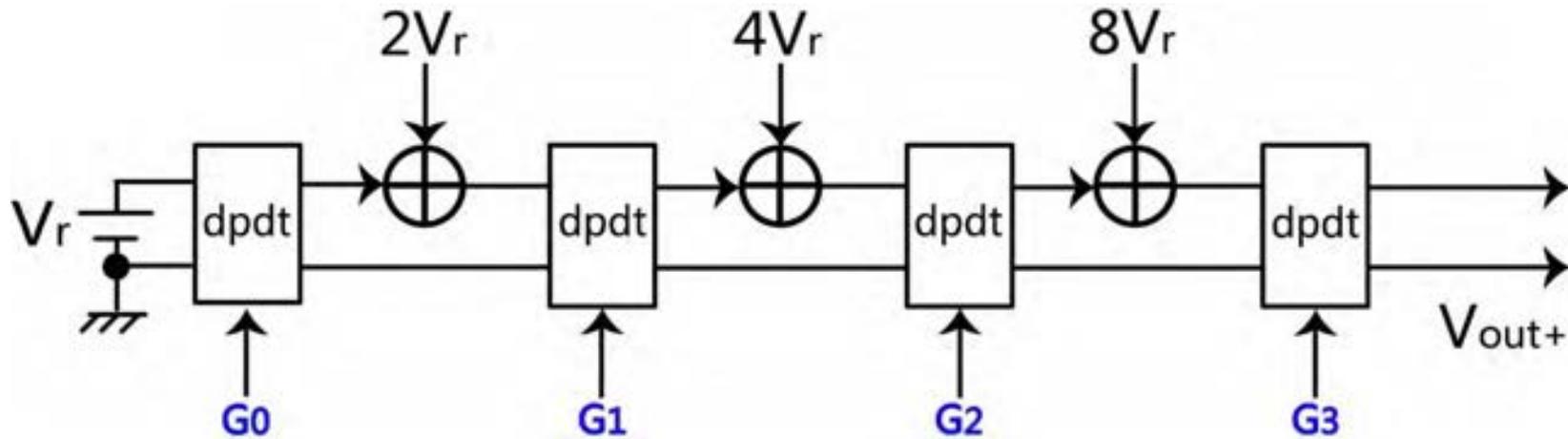


Gray-code入力のC-DACの動作 (データ=5の場合)

eg. Data=5



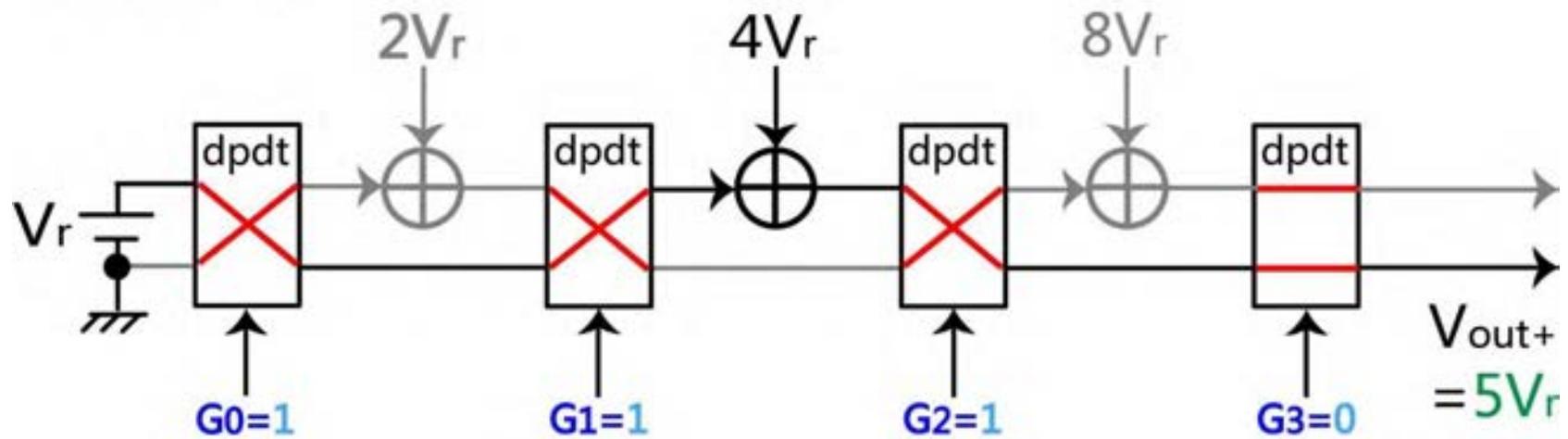
3. Gray-code入力の電圧加算型DAC



Gray-code入力の電圧加算型DAC

Gray-code入力のV-DACの動作 (データ=5の場合)

eg. Data=5



目次

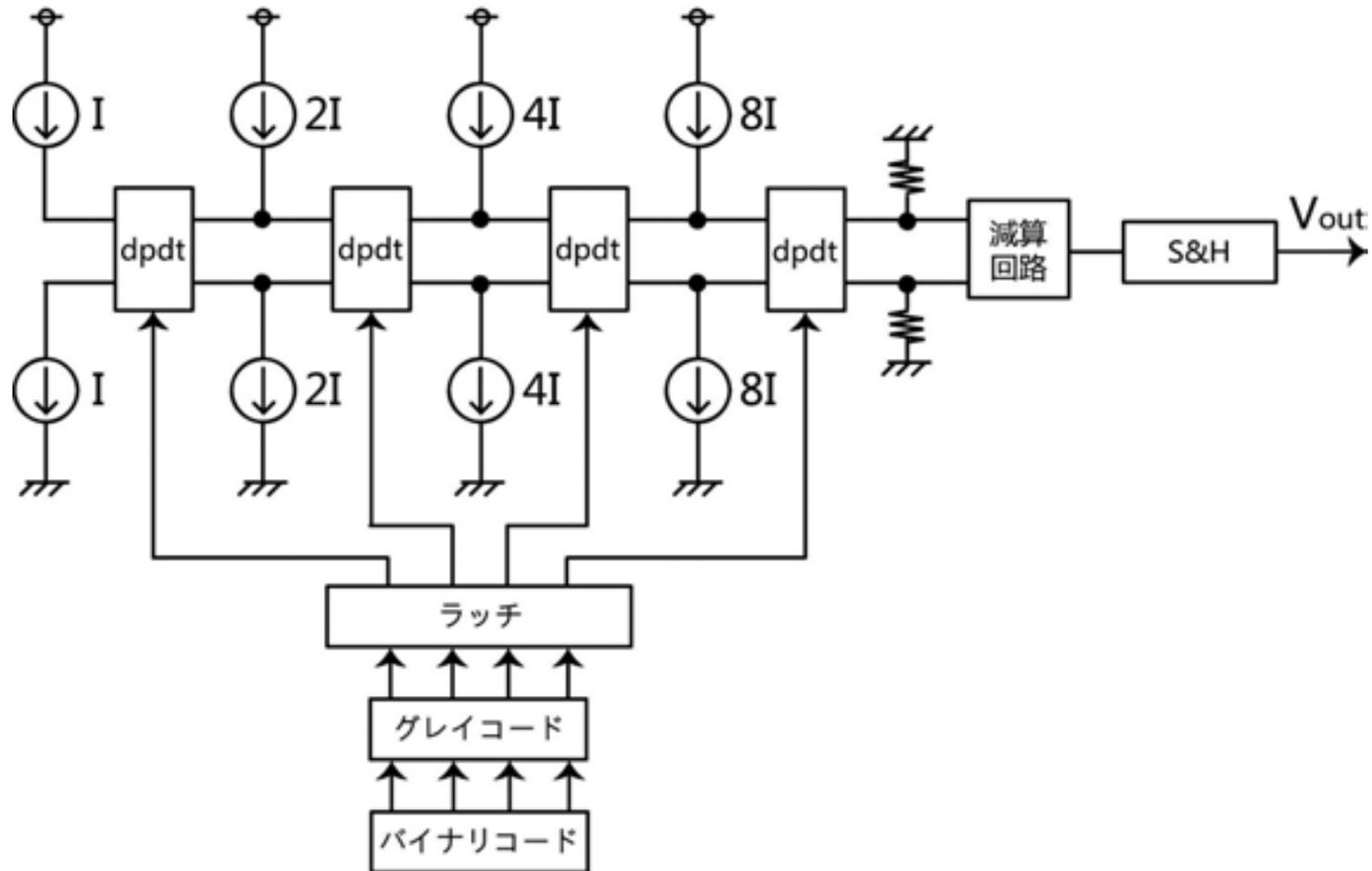
- I. 研究背景・目的
- II. 提案するGray-code入力のDACの構成と動作
- III. SPICEによるシミュレーション検証
- IV. まとめ

SPICEによるシミュレーション検証

1. Gray-code入力の電流出力型DAC (I-DAC) のシミュレーション
2. Gray-code入力の容量型DAC (C-DAC) のシミュレーション
3. Gray-code入力の電圧加算型DAC (V-DAC) のシミュレーション
4. グリッチ除去の検証

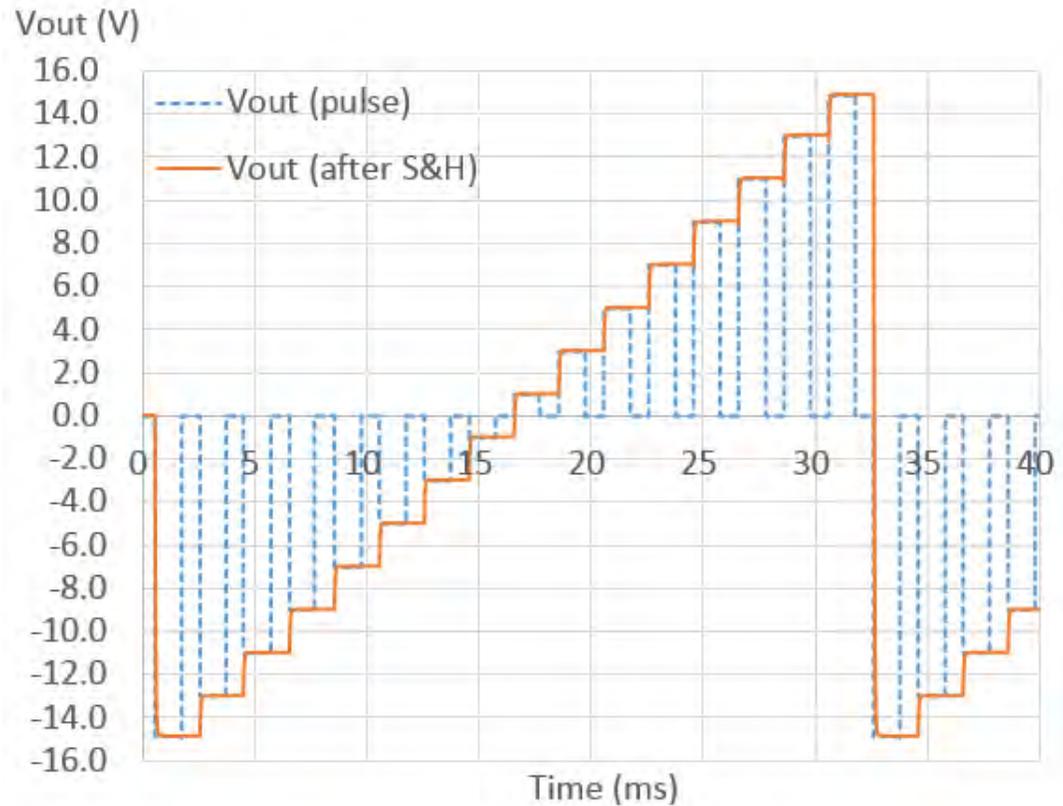
* LTspice IVを使用

1. Gray-code入力のI-DACのSPICE実現

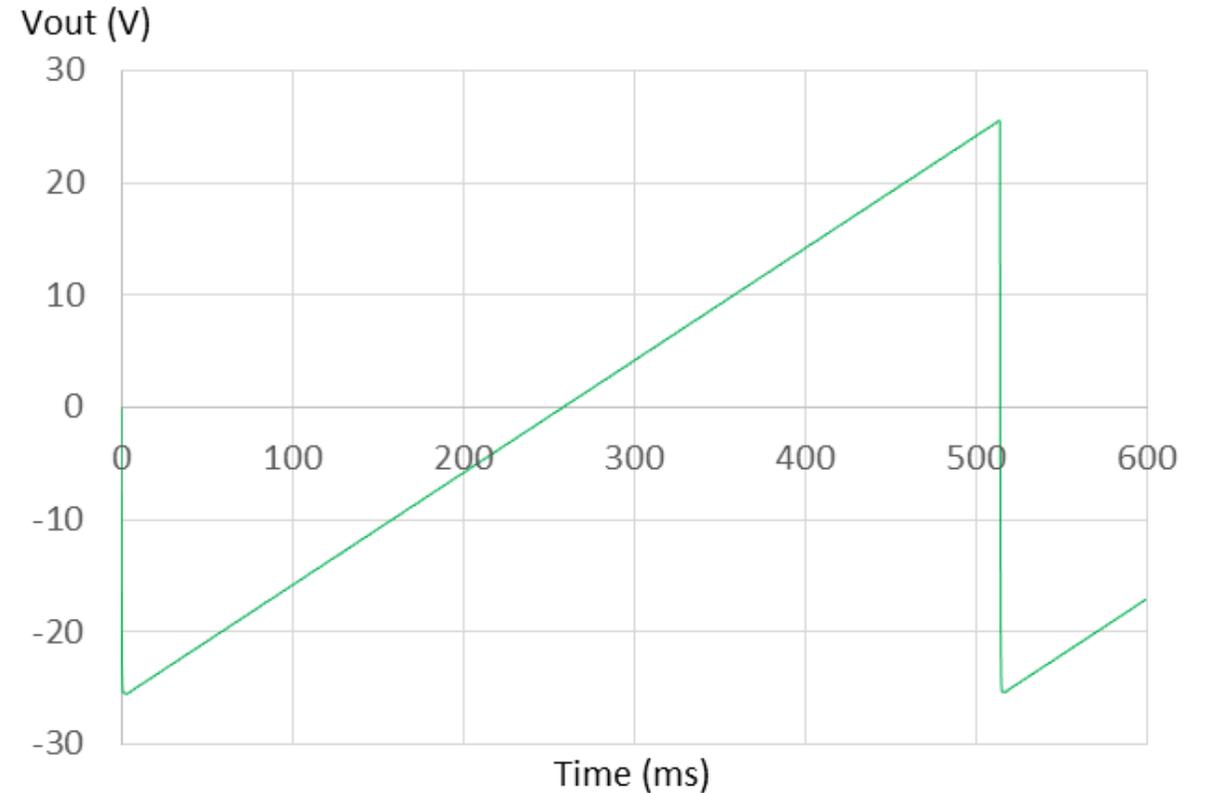


Gray-code入力の電流出力型DAC

1. Gray-code入力のI-DACのシミュレーション

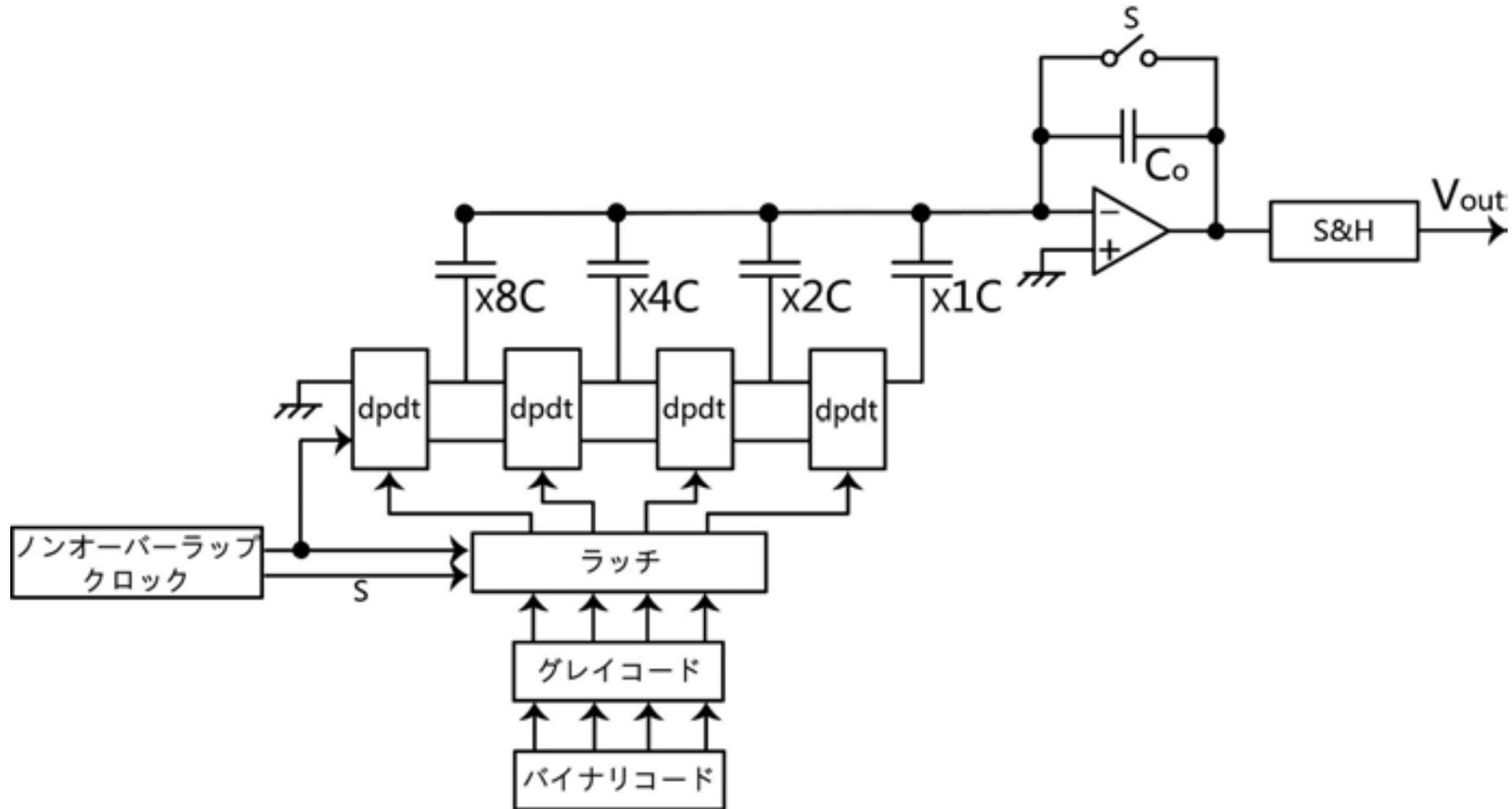


4bit I-DAC



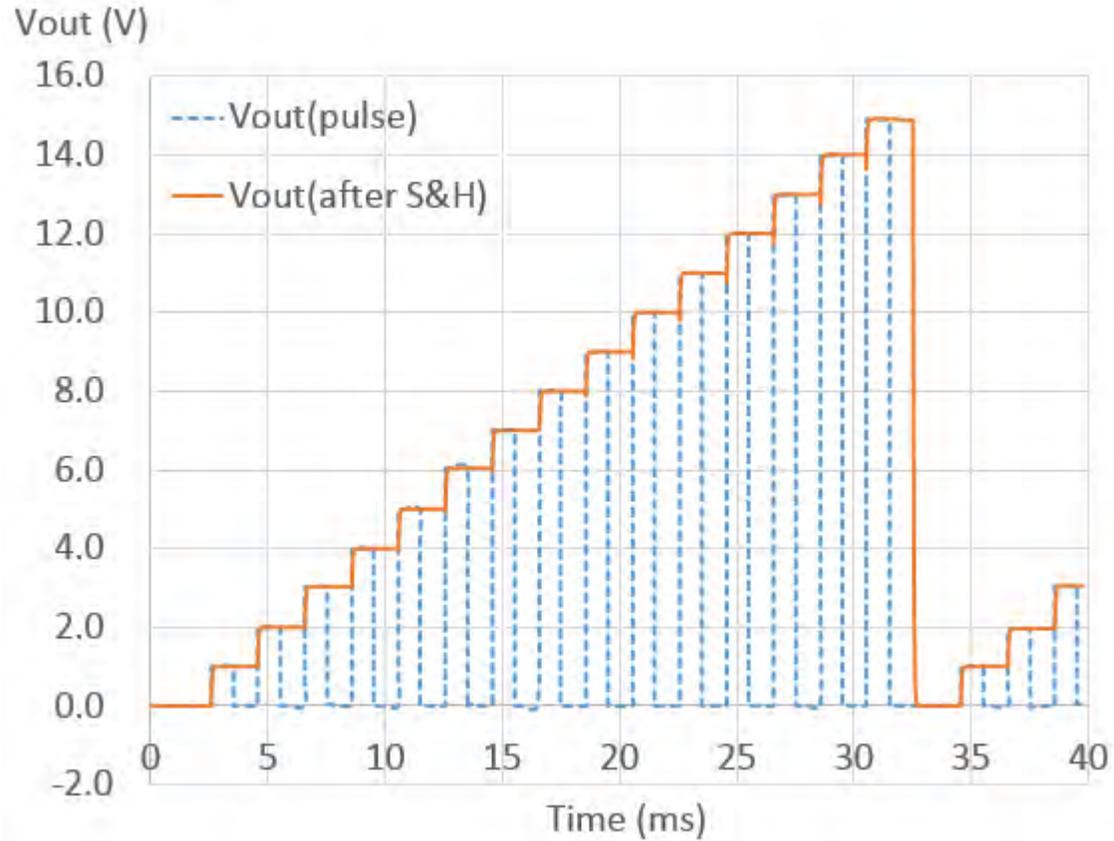
8bit I-DAC

2. Gray-code入力のC-DACのSPICE実現

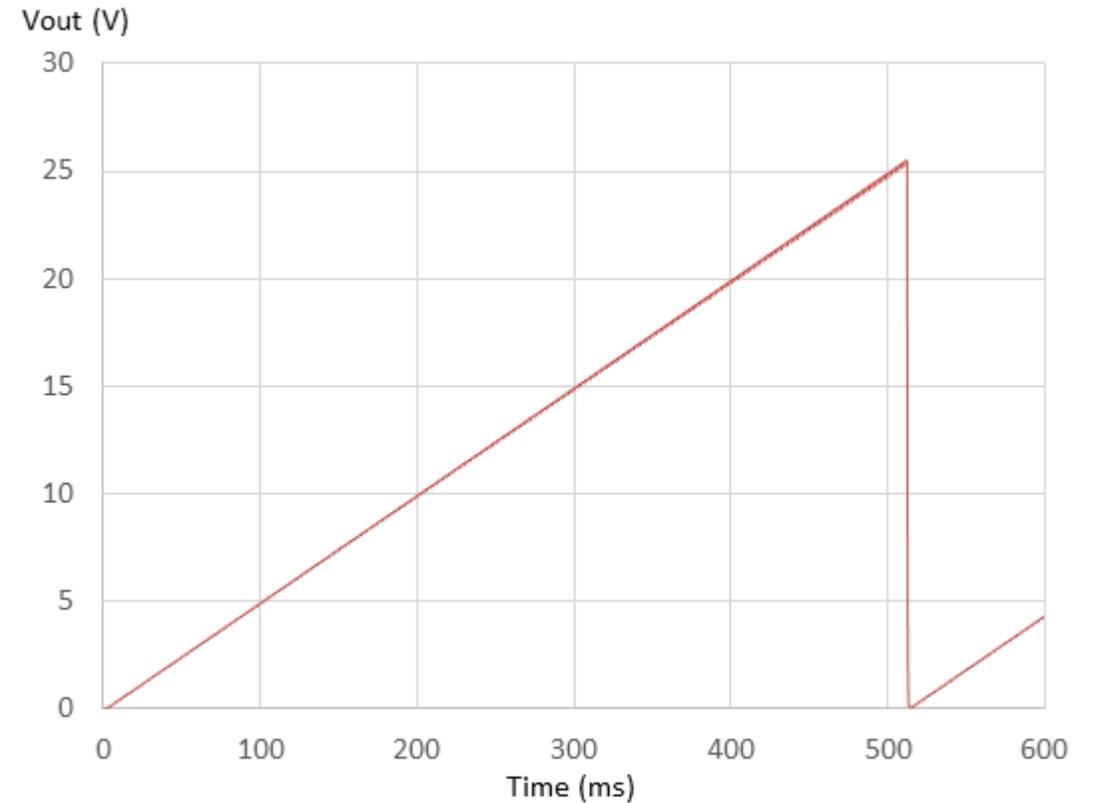


Gray-code入力の容量型DAC

2. Gray-code入力のC-DACのシミュレーション

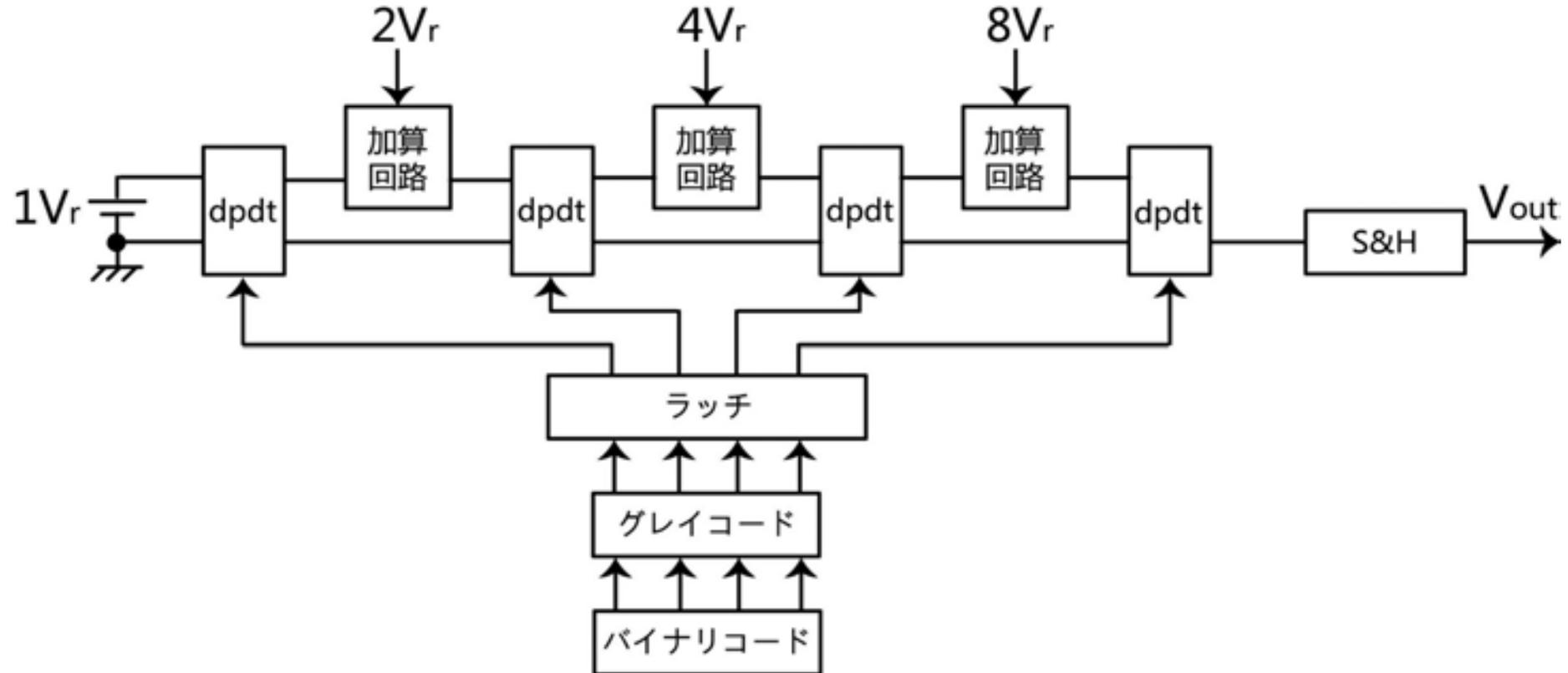


4bit C-DAC



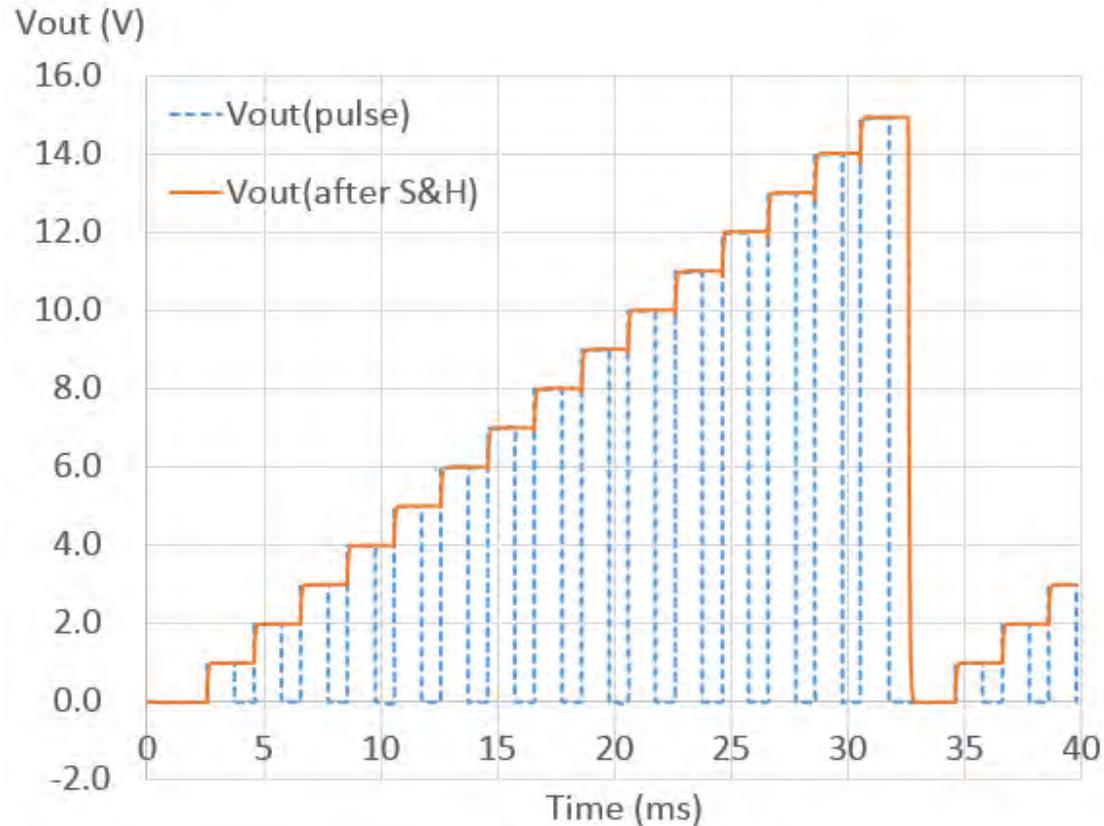
8bit C-DAC

3. Gray-code入力のV-DACのSPICE実現

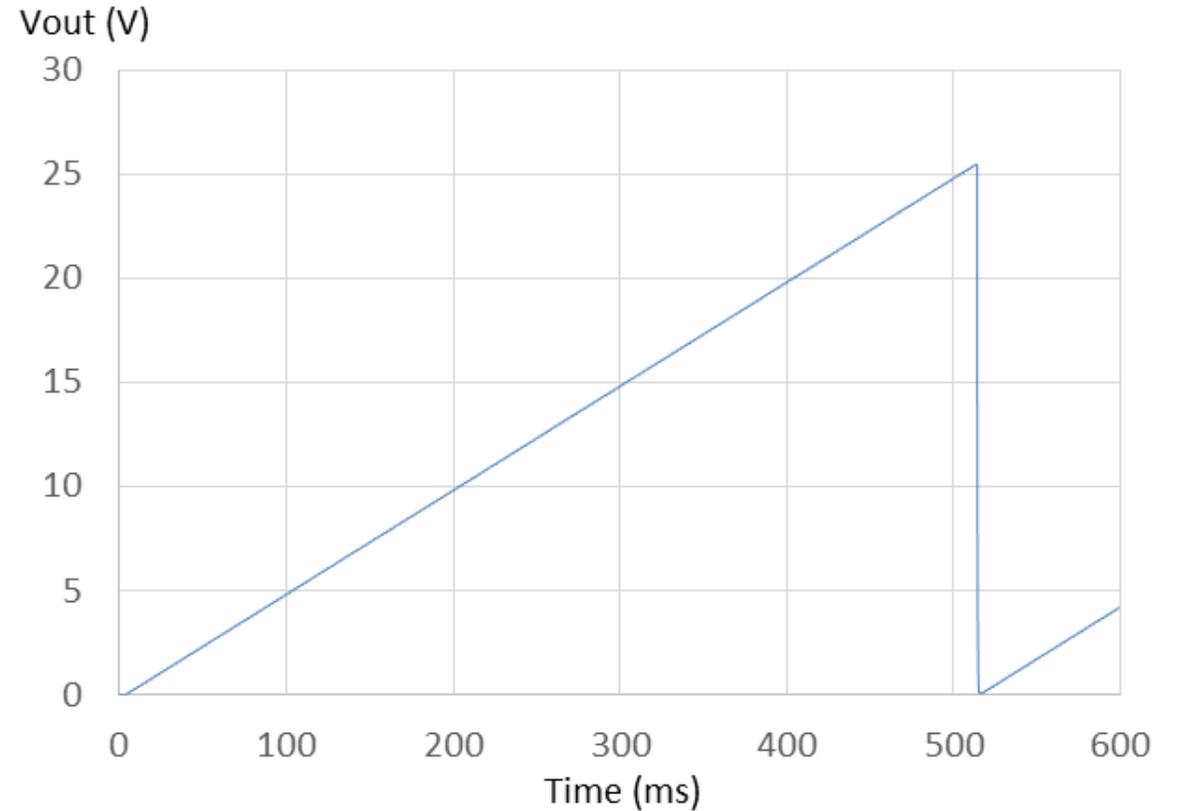


Gray-code入力の電圧加算型DAC

3. Gray-code入力のV-DACのシミュレーション

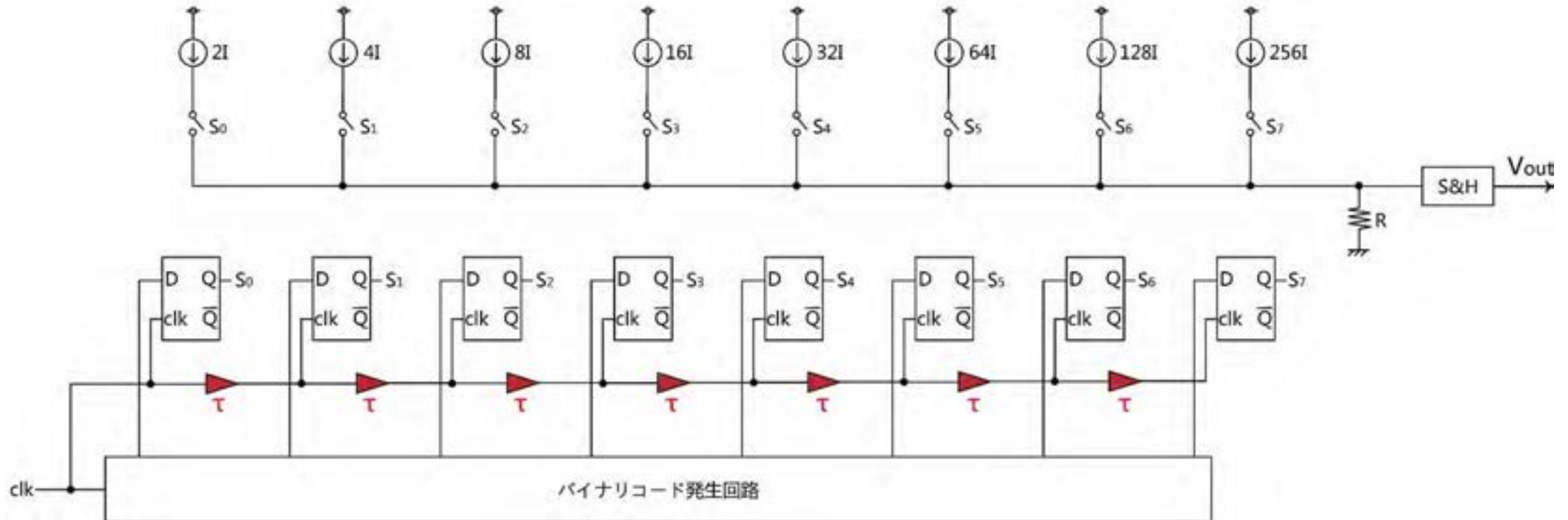


4bit V-DAC



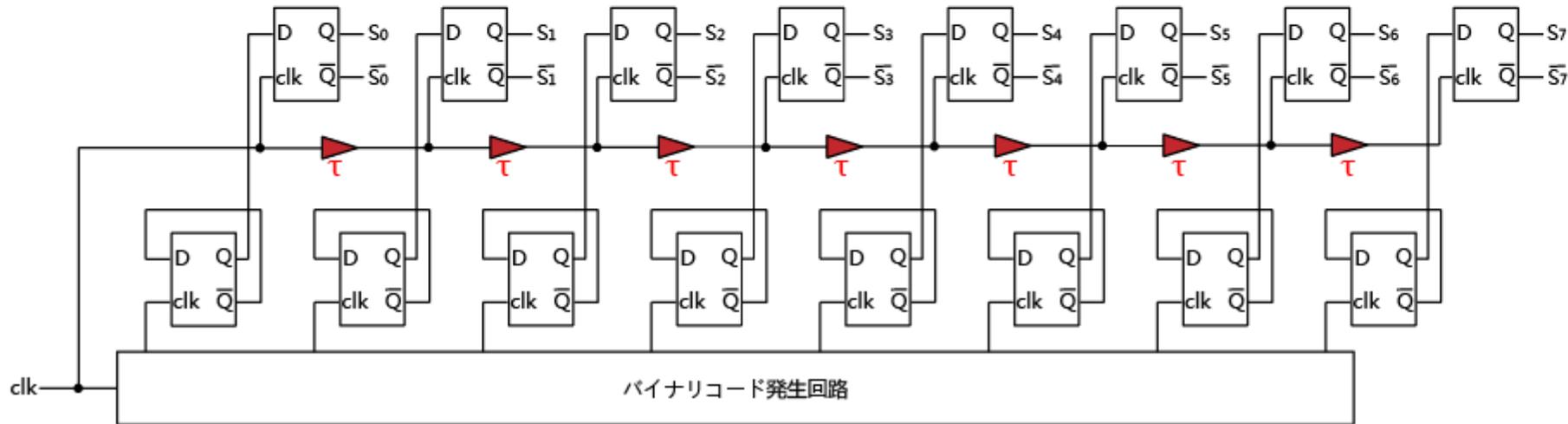
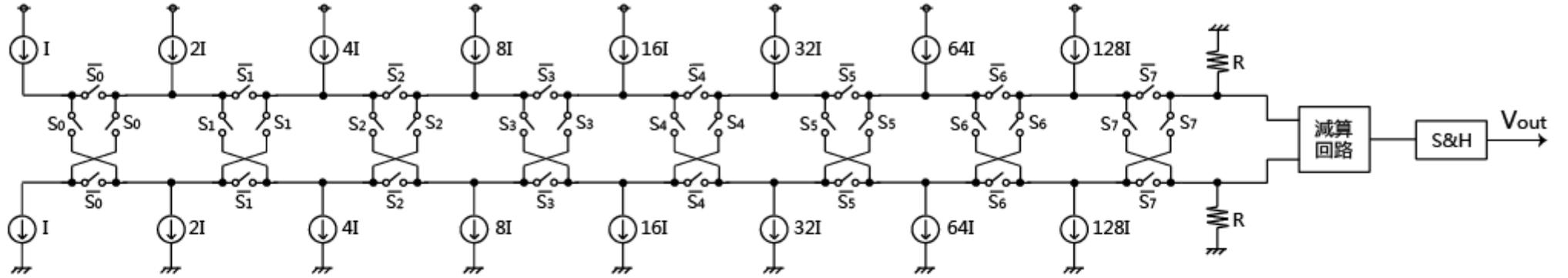
8bit V-DAC

4. グリッチ除去の検証



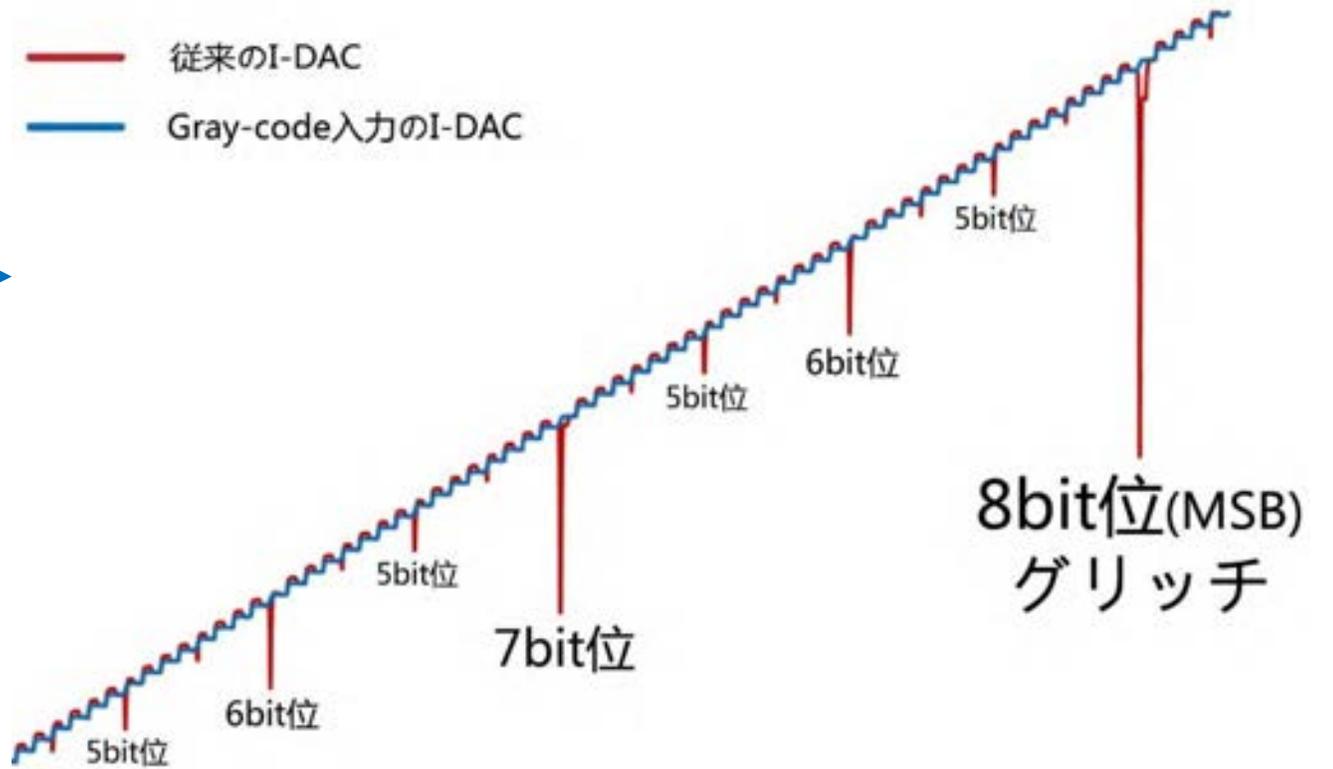
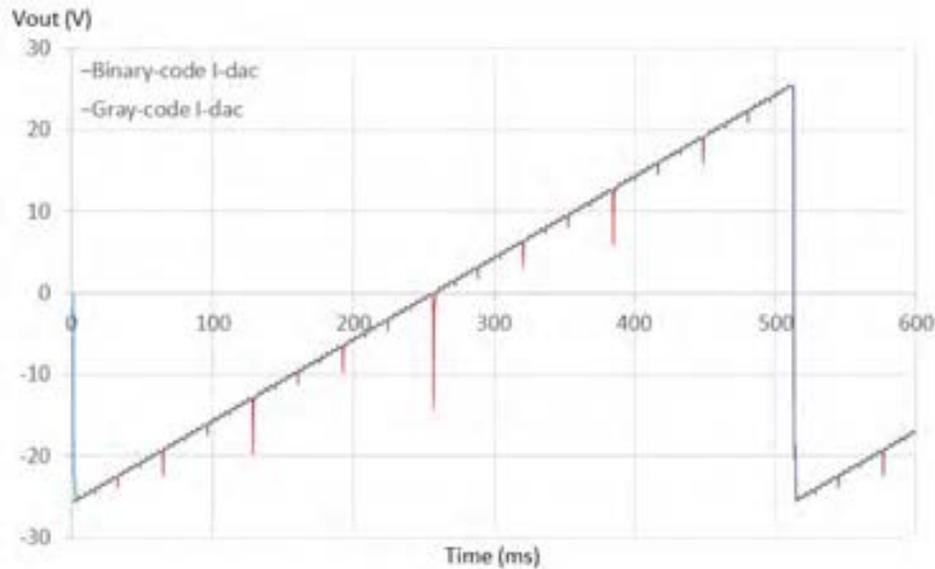
スイッチング遅延がついた従来のI-DAC (8 bit)

4. グリッチ除去の検証



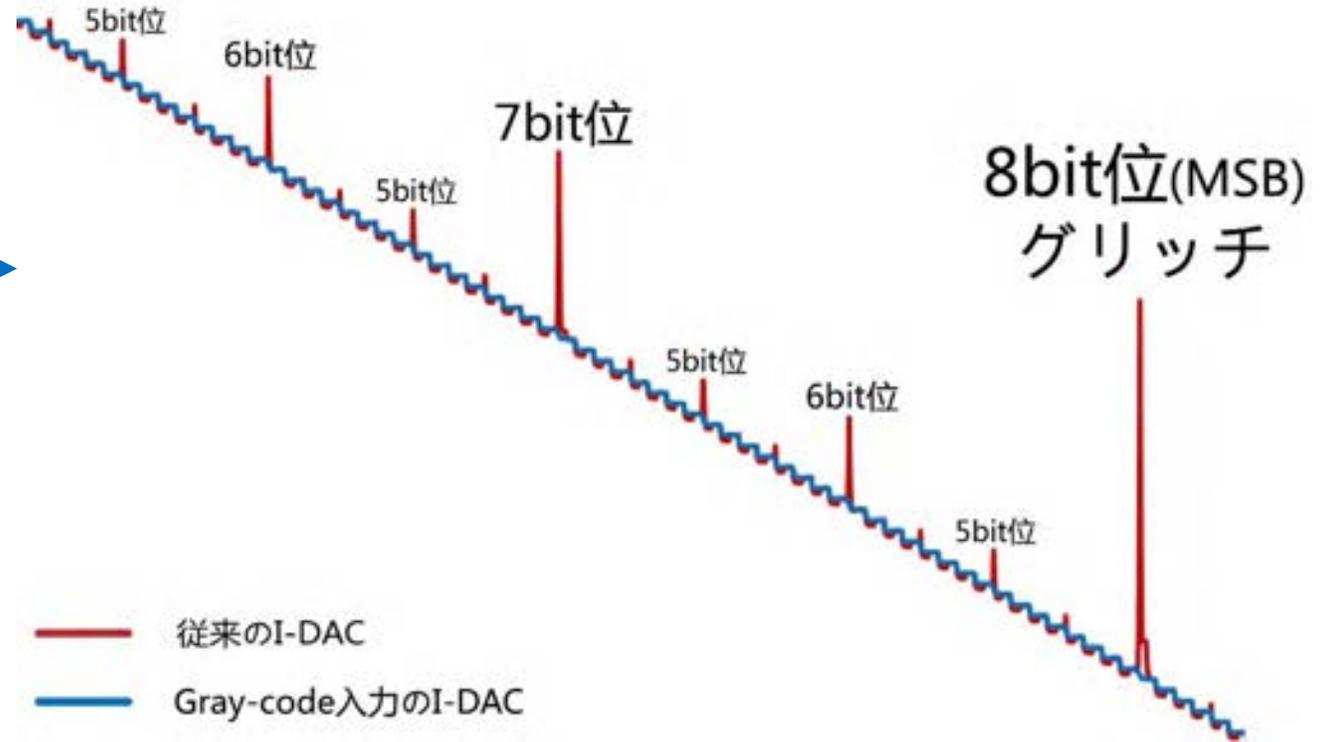
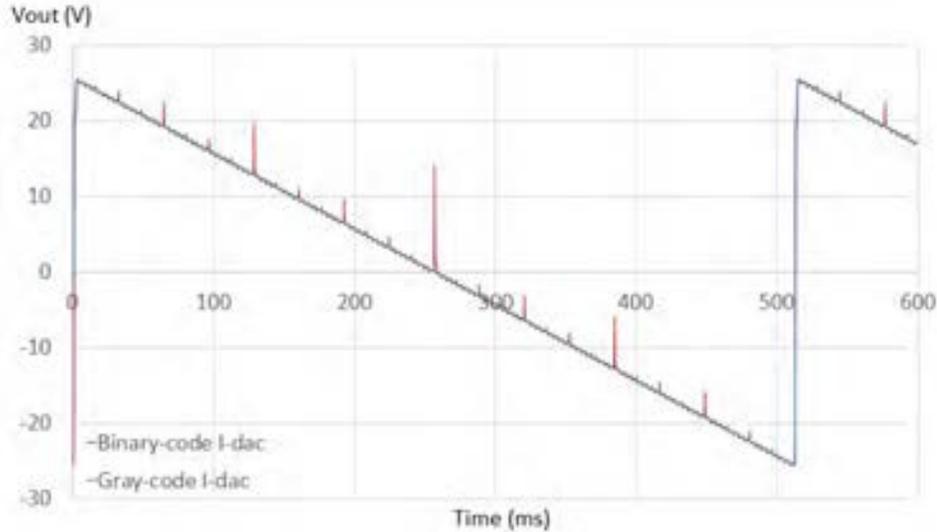
スイッチング遅延が**ついた**Gray-code入力のI-DAC (8 bit)

アップスウィーピング



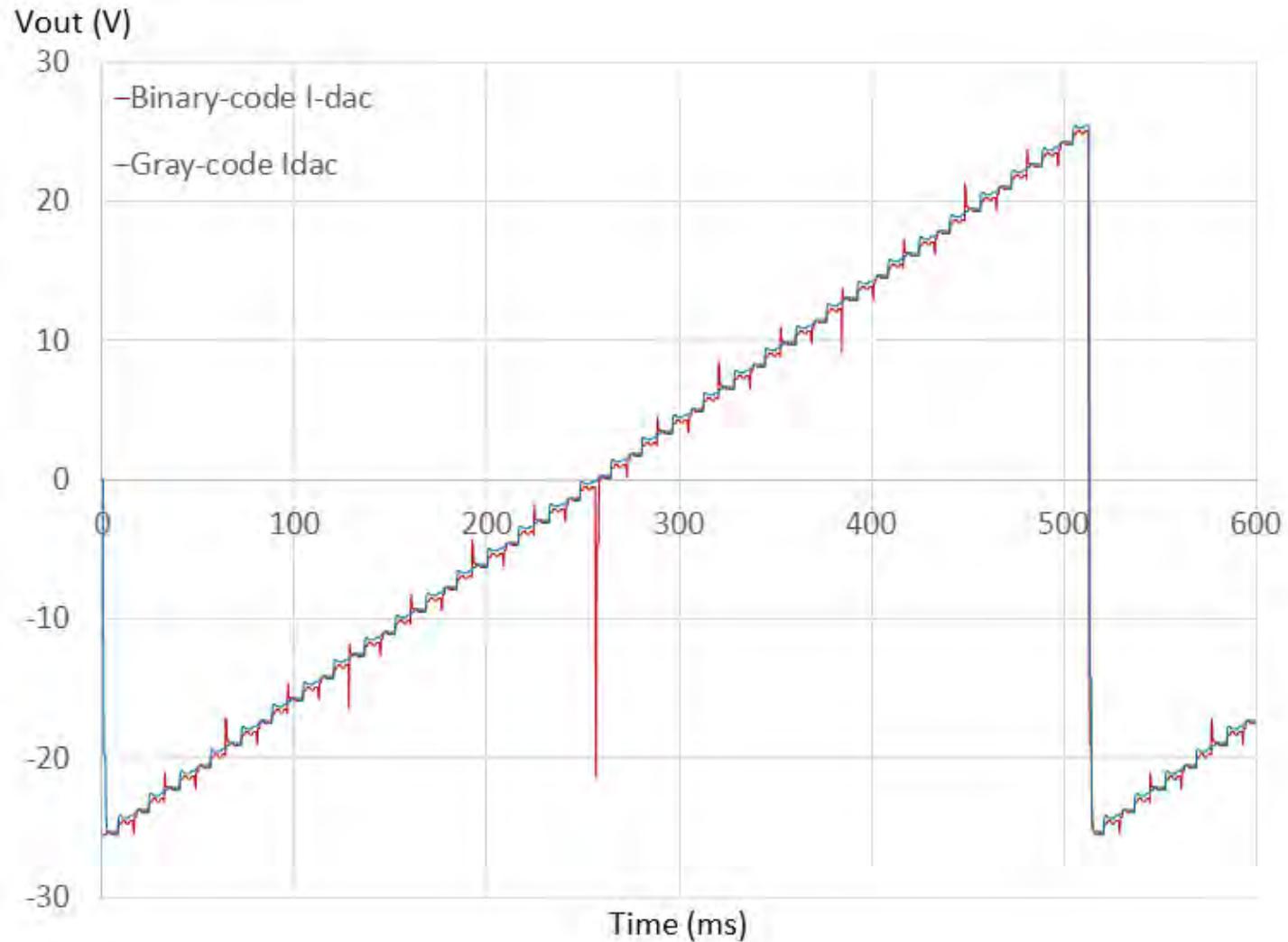
従来のI-DAC vs. Gray-code入力のI-DAC

ダウンスウィーピング



従来のI-DAC vs. Gray-code入力のI-DAC

ランダムなスイッチング遅延

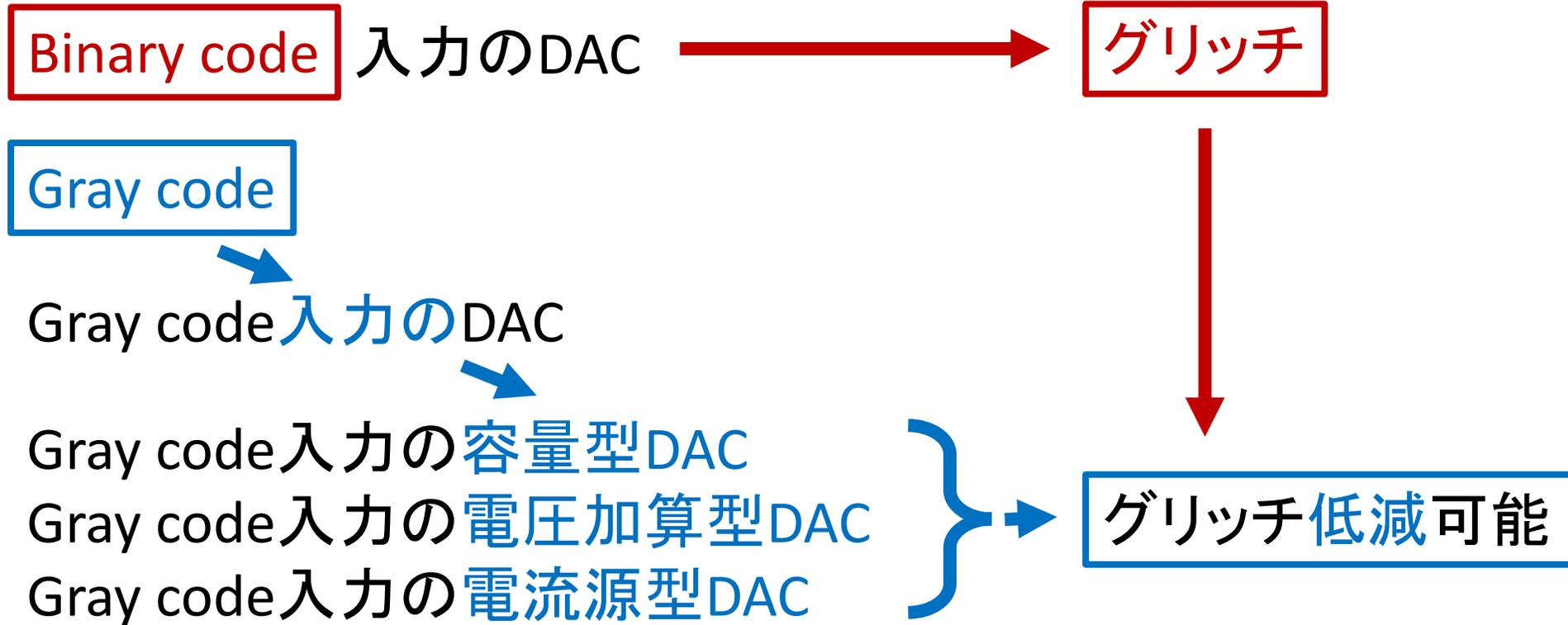


従来のI-DAC vs. Gray-code入力のI-DAC

目次

- I. 研究背景・目的
- II. 提案するGray-code入力のDACの構成と動作
- III. SPICEによるシミュレーション検証
- IV. まとめ

まとめ



今後の課題:

電圧/電流スイッチマトリックスのMOSFETでの設計

Q/A

Q:この研究は先行研究に比べて発展された点は何ですか。

A:研究背景でお話したように、Gray-codeを入力としての綺麗な構成のDACは実現が難しいと考えられています。本論文ではGray-code入力のDACが実現できることを示します。