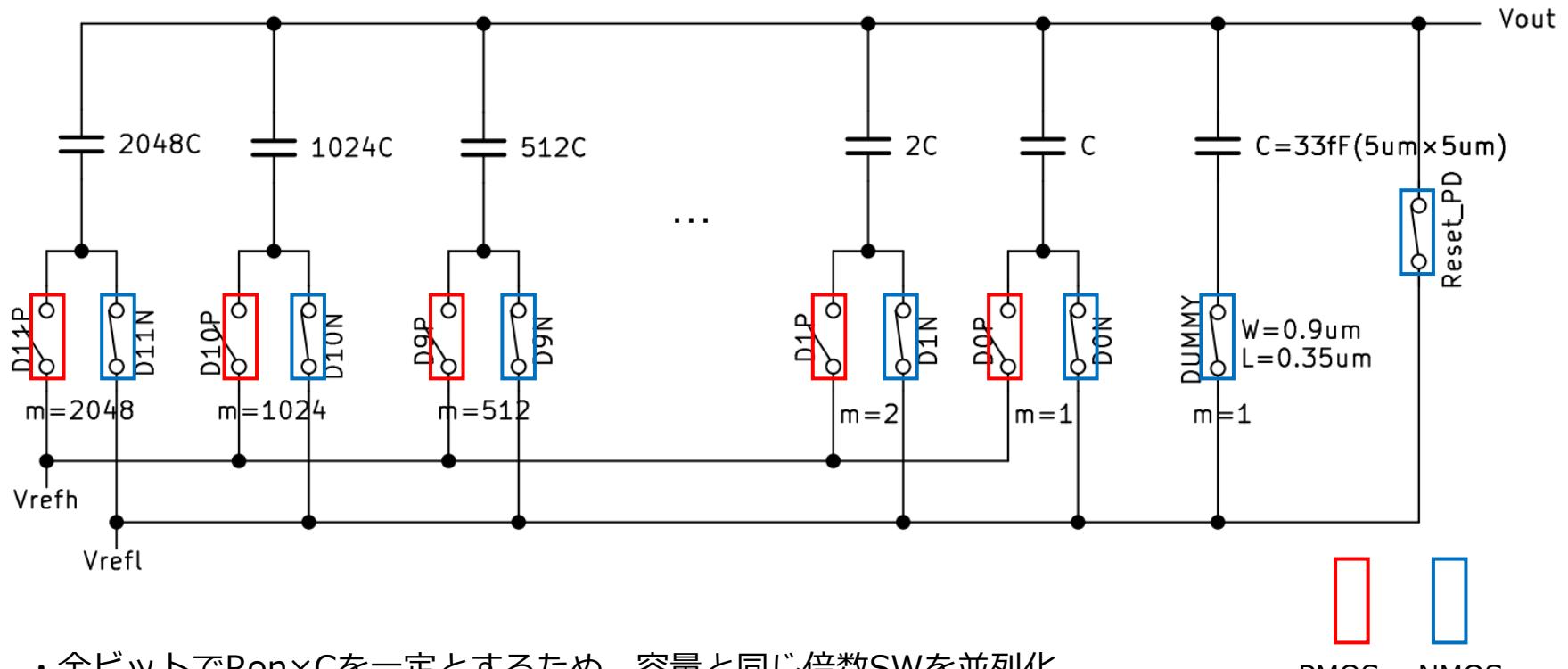


12bitキャパシタ型DAC 動作説明

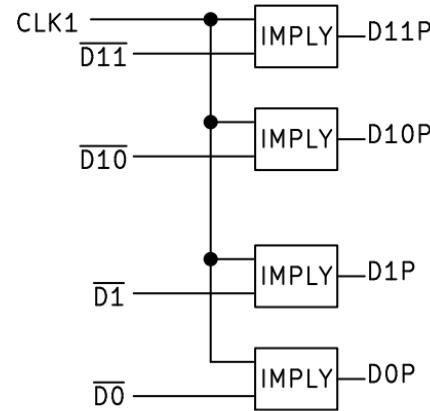
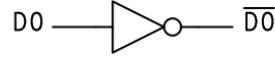
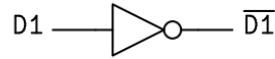
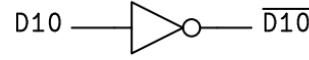
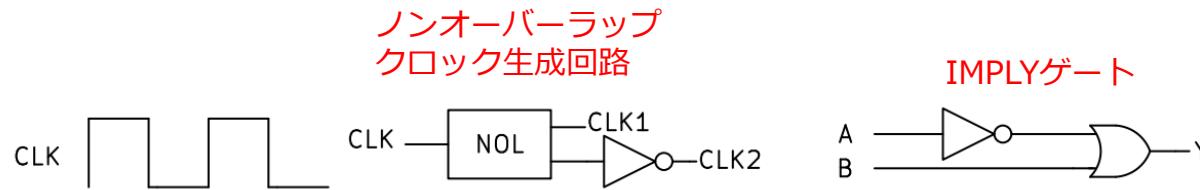
野田慶太

CDACアナログ部構成



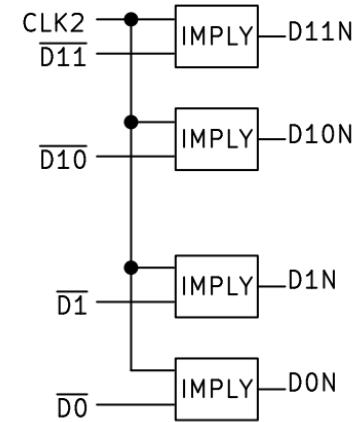
- 全ビットで $Ron \times C$ を一定とするため、容量と同じ倍数SWを並列化
- SWのサイズは全て同じ
- DUMMYのスイッチは常時ON

CDACデジタル部構成



デジタル入力を反転

PMOSスイッチの制御信号生成



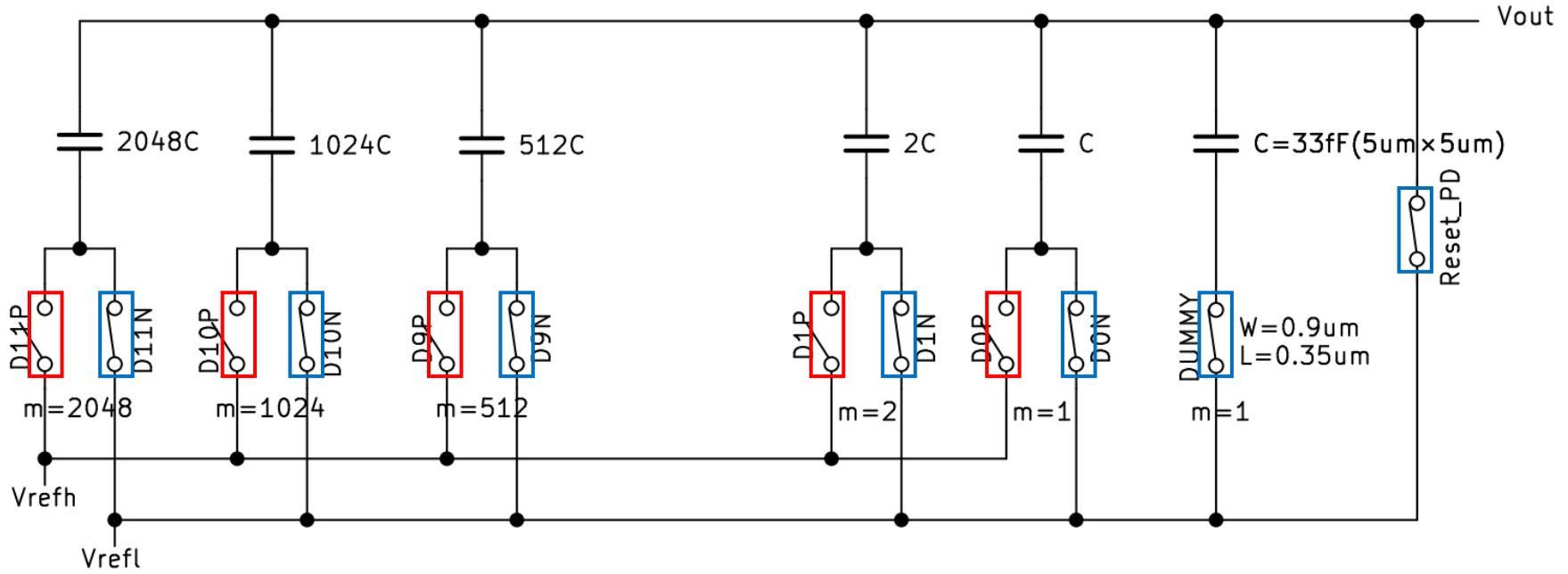
NMOSスイッチの制御信号生成

PMOSスイッチ:CLK1がHighかつDがHighのときのみON

NMOSスイッチ:CLK2がLowのとき(リセット中)、またはDがLowのときON



動作原理

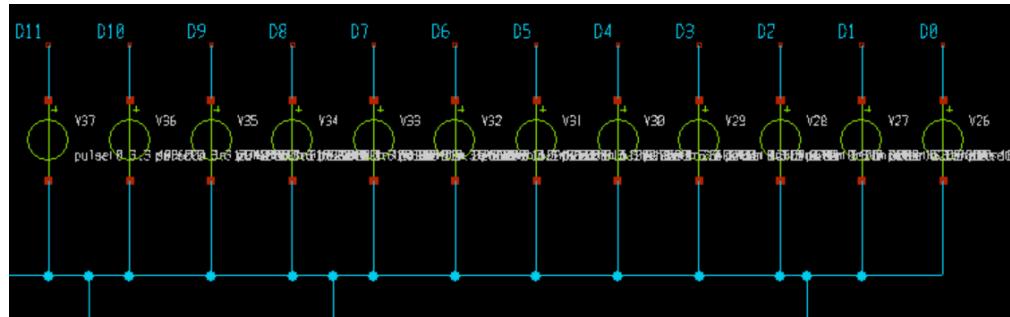


リセット期間:NMOSスイッチを全てONにする→Cの電位差が無くなる→電荷リセット
 ↑繰り返し

電荷再配分期間:Reset_PDはOFF、デジタル値がHighのときはPMOS、LowのときはNMOSがONになり、Vrefと繋がる→電荷保存則により、Voutが出力

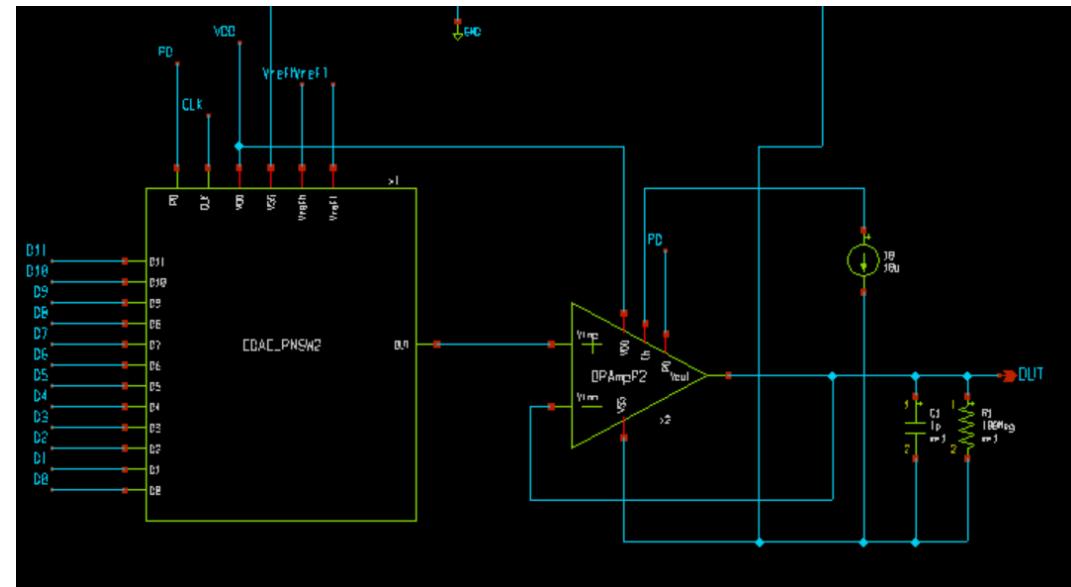
PDがLowのときは、常にリセットスイッチがONとなるためVoutはVreflで一定

DNL/INLの測定方法



000000000000→111111111111
まで、全通りのデジタル値を入力

LSB(D0)はCLKの2倍、D1はその2倍とい
うように、上位に行くごとに周期を2倍に
していくと全通り入力が可能



OUTのデータを取得する
ngspice で
wrdata ファイル名.txt(or csv) v(OUT)
と入力

Xschem/simulations/userConfの中に
出力される

DNL/INLの測定方法

```
import numpy as np
import matplotlib.pyplot as plt
import os

# =====
# ユーザー設定エリア
# =====
TARGET_FILENAME = "extracted_test5.txt"

# DAC設定
RESOLUTION_BITS = 12

# サンプリング設定 (単位: 秒)
T_START = 190e-9 # 1点目
T_PERIOD = 200e-9 # 周期
# =====
```

ファイル名や分解能、サンプリング周期を設定

サンプリング周期は、最初の一点の位置を指定し、そこから等間隔でサンプリングする

ファイルサイズが大きいので、サンプリングしたデータのみを出力する機能
(ファイル名にextractedとつく)
→元のデータは不要なら消す

DNL/INL

このように結果が出力される
下は、サンプリングしたデータ
をプロットしたもの

