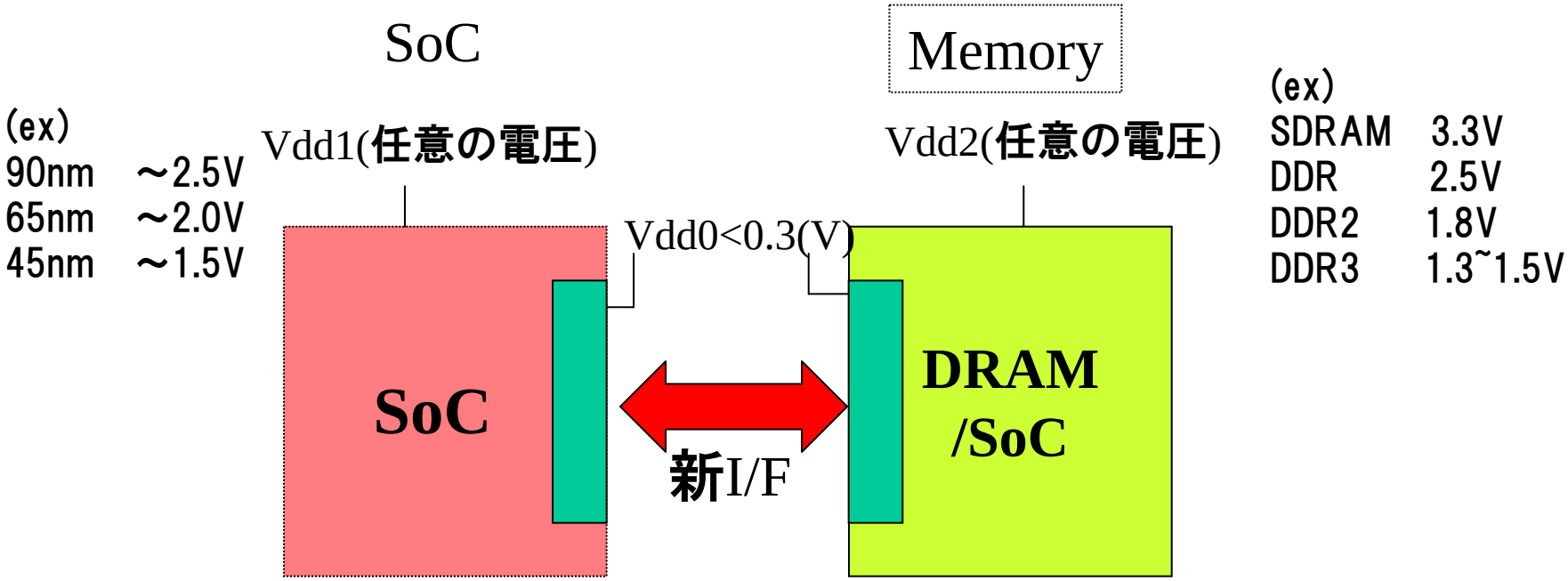
超低電力･超高速I/F技術
(特許取得済）

2015

Sorbus,Inc.

全体イメージ



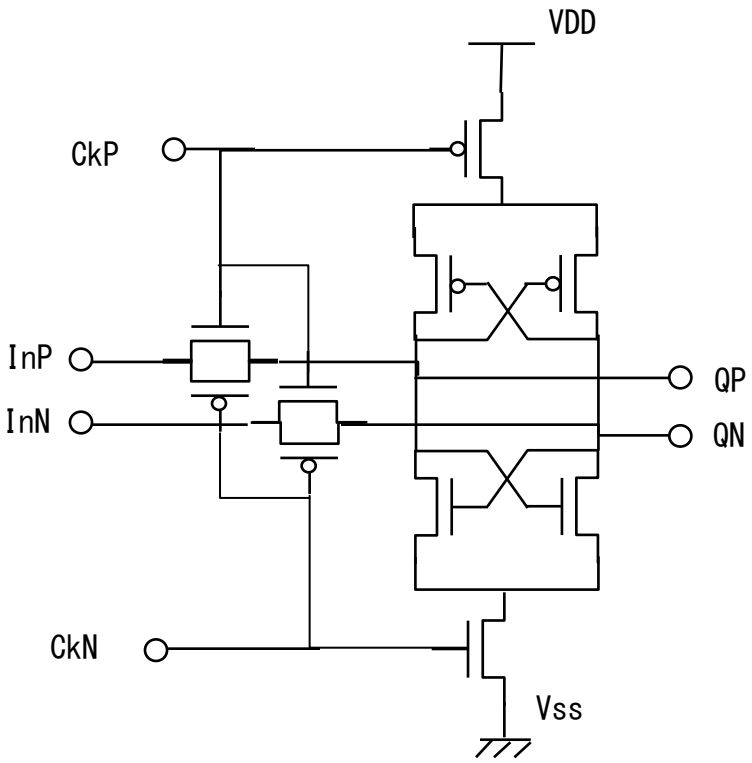
（１）Vdd1,Vdd2はﾌﾟﾛｾｽｼｭﾘﾝｸ見合いで下げてOK。
ｼｭﾘﾝｸはSoCとMemoryで連動する必要無し、それぞれ単独にｼｭﾘﾝｸ可能。

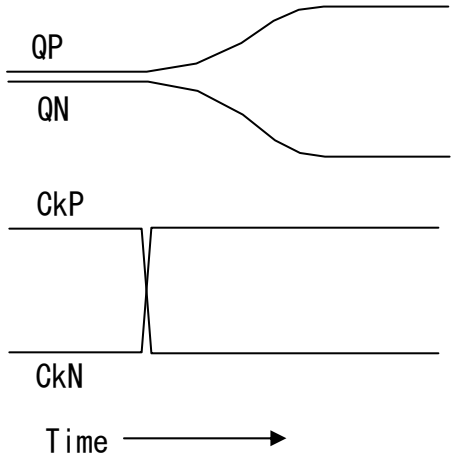
（２）消費電が大きい出力ﾄﾞﾗｲﾊﾞｰは電圧を下げて(0.2~0.3V)低電力化。高速化は低電圧対応の多値技術で対応。（特許取得済）

ラッチ型コンパレータのオフセット補正手段と多値復調への応用（特許取得済）

Sorbus, Inc.

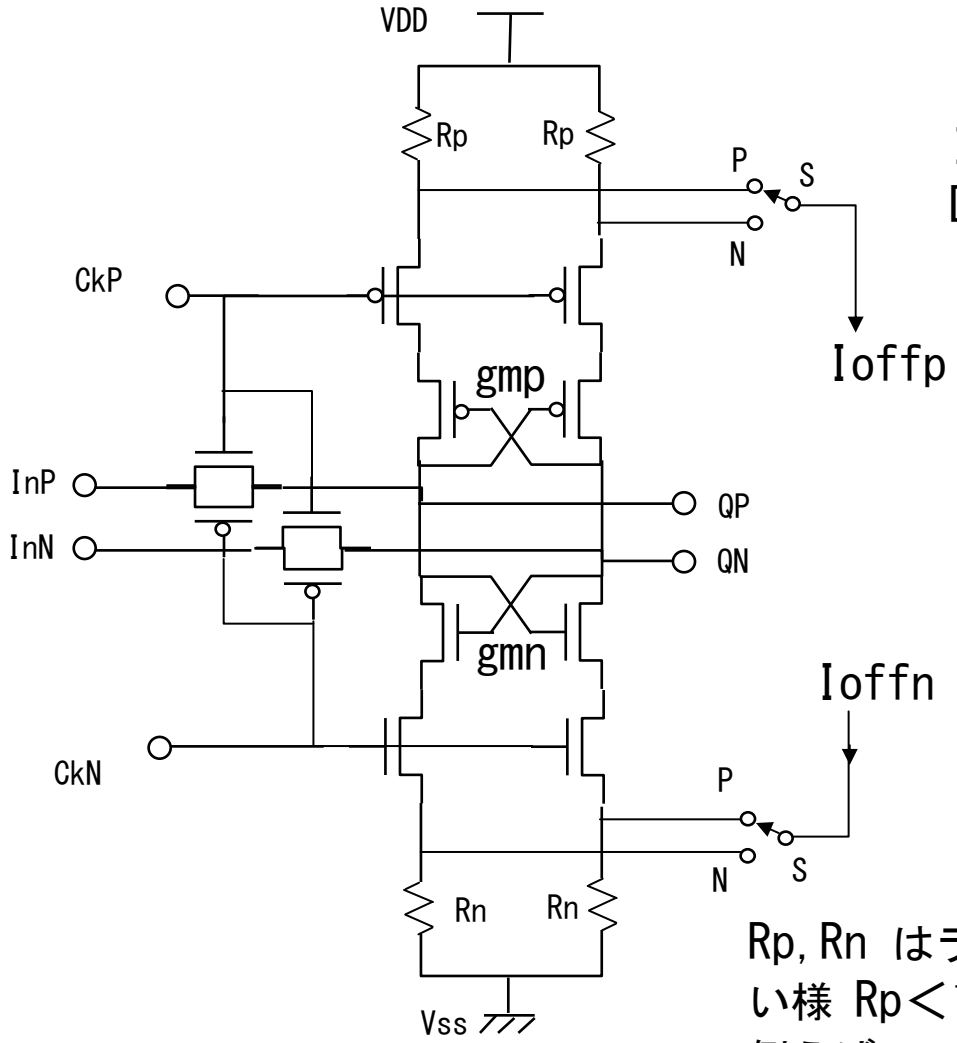
ラッチ型コンパレータの基本形





ラッチ型コンパレータの特徴
- アンプ型より高速動作が可能
- Trバラツキ等に起因するオフセット電圧を補正する事が困難

ラッチ型コンパレータのオフセット調整回路①



Ioffp, Ioffnは電流出力型DACより供給

オフセット補正量(入力換算)

S→P　Ioffp×Rp×kp
＋Ioffn×Rn×kn

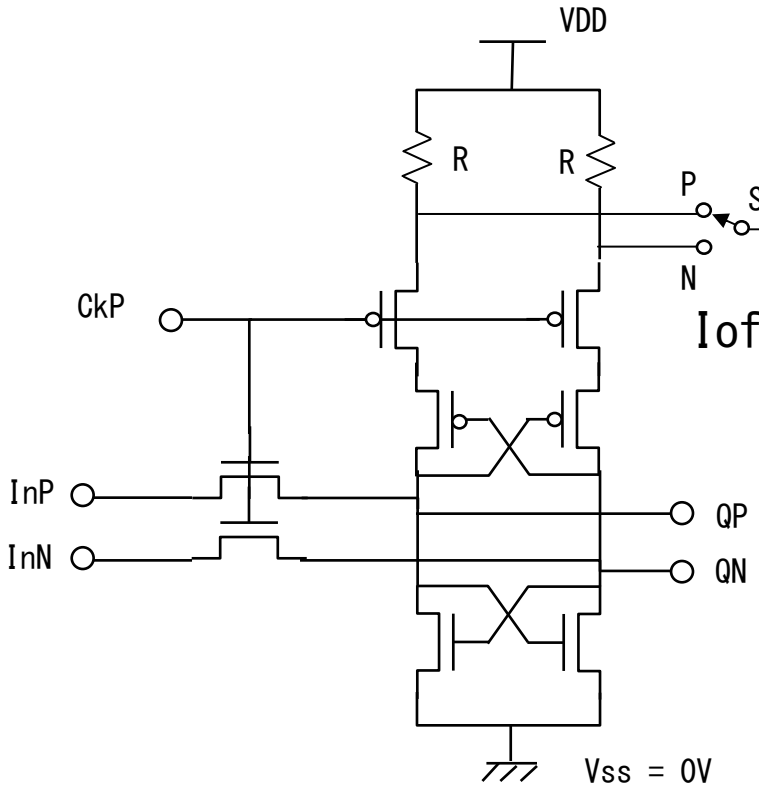
S→N　－Ioffp×Rp×kp
－Ioffn×Rn×kn

kp = gmp/(gmp+gmn)
Kn = gmn/(gmp+gmn)

Rp, Rn はラッチの動作にあまり影響を与えない様 Rp＜1/gmp 、Rn＜1/gmn 程度に選ぶ。
例えばgmp=0.5mS, gmn= 1mS ならRp=700Ω
Rn= 350Ω程度

ラッチ型コンパレータのオフセット調整回路②

入力電圧範囲が　≦Vthn（:Nch-Vth）の場合は回路が簡略化出来る



オフセット補正量

S→P　Ioff×R

S→N　－Ioff×R

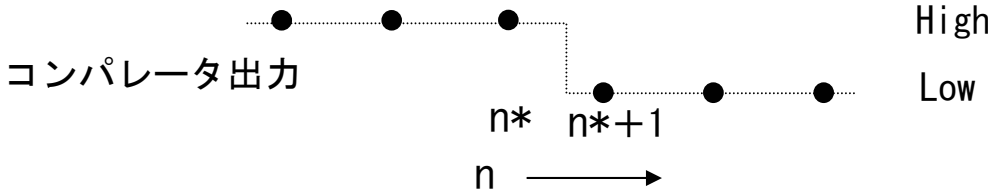
オフセット補正用の電流出力DACのビット数をMとすれば、極性スイッチSと併せて n・⊿V（ -(2^M-1)≦n(整数)≦2^M-1, ⊿V= Io・R)のオフセット補正が可能となる。ここで、IoはDACの電流量子。

入力電圧範囲が　≧VDD＋Vthp（:Pch-Vth,負）の場合は上記回路でNch→Pch, Pch→Nch,VDD→Vss,Vss→VDDにそれぞれ置き換えて使う事が出来る

ラッチ型コンパレータのオフセット調整方法

1. ２つの入力端子を同電位にする
2. オフセット調整のｎ値を変えてコンパレータの出力が変化する点を探す
3. 上記変化点のnを、n*とn*＋1とすると、オフセット調整のｎ値をこのn*又はn*＋１として調整を終了する。最終的なオフセット値は、
   －⊿V ≦オフセット値≦ ⊿V　　（⊿V= Io・R, Io:DACの電流量子）
   に収まる事となる。

⊿Vを2mV,DACのビット数を5ビットとすれば、－62mV～＋62mVのオフセットを±2mV以内に補正が可能となる。



オフセット補正機能付きラッチ型コンパレータの多値復調への応用（拡張）

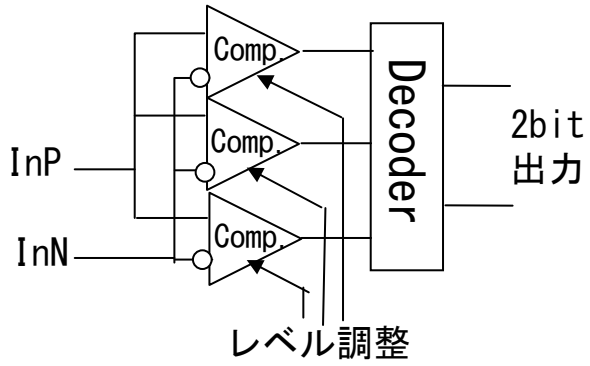
高速Chip間インタフェース等において差動かつ多値変調技術が注目されているが、この復調に前記オフセット補正機能付きのラッチ型コンパレータを応用する事により高速の多値復調が可能となる。
ここでは前記オフセット補正機能を拡張して、多値比較レベルの設定とオフセット補正を同時に行う事が可能となる。

4値変調の例（InP, InNはそれぞれ差動の正相、逆相入力）

Signal_HH	Signal_HL	Signal_LH	Signal_LL
InP ——— +150mV	InP ——— +75mV	InN ——— +75mV	InN ——— +150mV
InN ——— 0mV	InN ——— +25mV	InP ——— +25mV	InP ——— 0mV

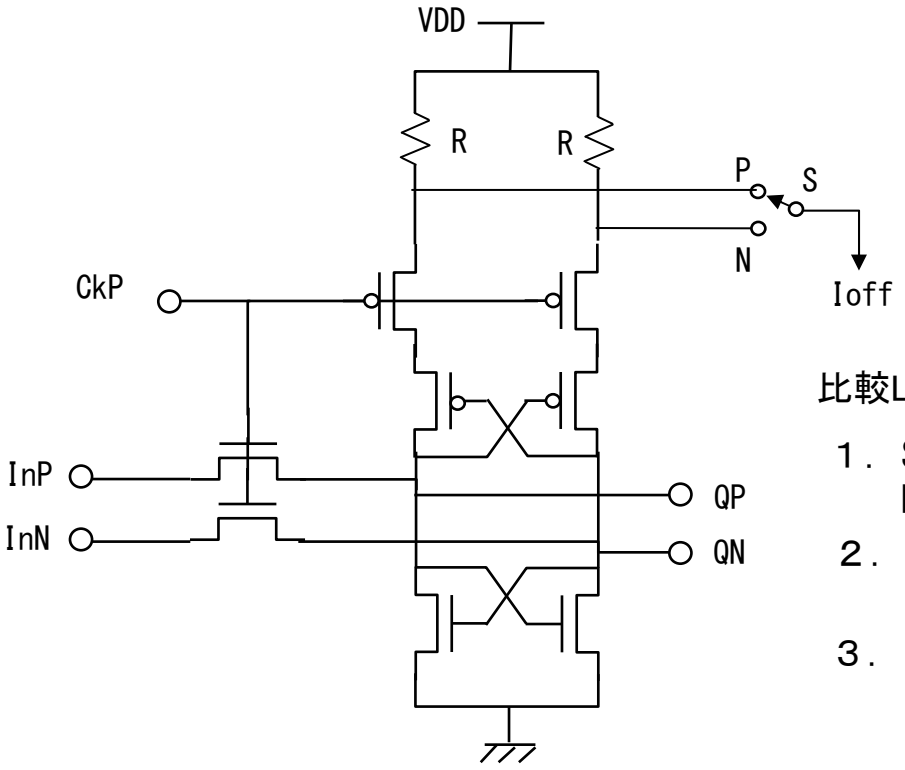
4値復調には比較レベルが異なる３つのコンパレータが必要

	InP－InN	Compare Level	
Signal_HH	+150mV		
		+100mV	Level_1
Signal_HL	+50mV		
		0mV	Level_0
Signal_LH	−50mV		
		−100mV	Level_/1
Signal_LL	−150mV		



差動4値入力を復調する３つのコンパレータの比較レベル(オフセット)調整

オフセット調整範囲を拡大して比較レベルの設定に応用する



オフセット調整レベル＝n・⊿V

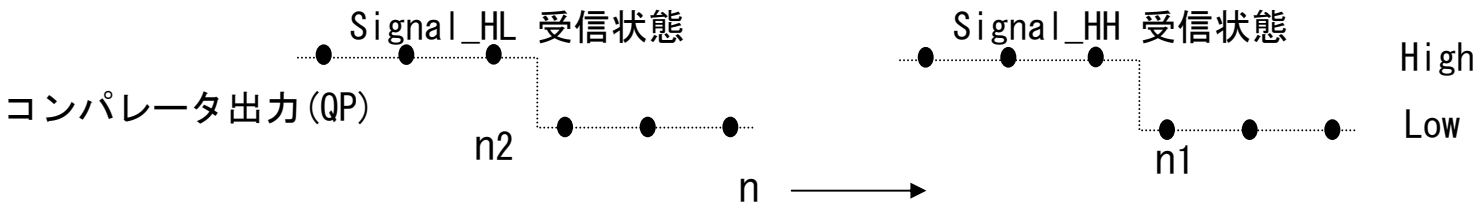
⊿V＝ IO・R

-(2^M−1)≦ n(整数) ≦ 2^M−1

⊿V= 8mV, M = 5bitとすれば±248mVの範囲でオフセットが調整可能

比較Level_1 の設定手順

1. Signal_HH 受信状態でコンパレータ出力がLowとなる最小のnを見つけこれをn1とする。
2. Signal_HL 受信状態でコンパレータ出力がHighとなる最大のnを見つけこれをn2とする。
3. n_1＝(n1+n2)/2 として比較レベルLevel_1の設定を完了



差動4値入力を復調する３つのコンパレータの比較レベル(オフセット)調整（続）

比較Level_0 の設定手順

１．Signal_HL 受信状態でコンパレータ出力がLowとなる最小のnを見つけこれをn1とする。
２．Signal_LH 受信状態でコンパレータ出力がHighとなる最大のnを見つけこれをn2とする。
３．n_0＝(n1+n2)/2 として比較レベルLevel_0の設定を完了

比較Level_/1 の設定手順

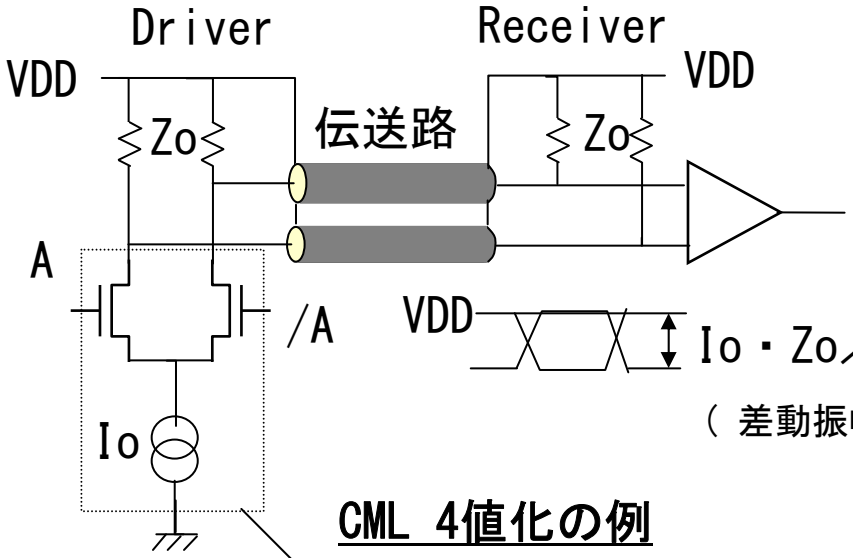
１．Signal_LH 受信状態でコンパレータ出力がLowとなる最小のnを見つけこれをn1とする。
２．Signal_LL 受信状態でコンパレータ出力がHighとなる最大のnを見つけこれをn2とする。
３．n_/1＝(n1+n2)/2 として比較レベルLevel_/1の設定を完了

超低電力高速インタフェース
多値変調回路
（特許取得済）

Sorbus,Inc.

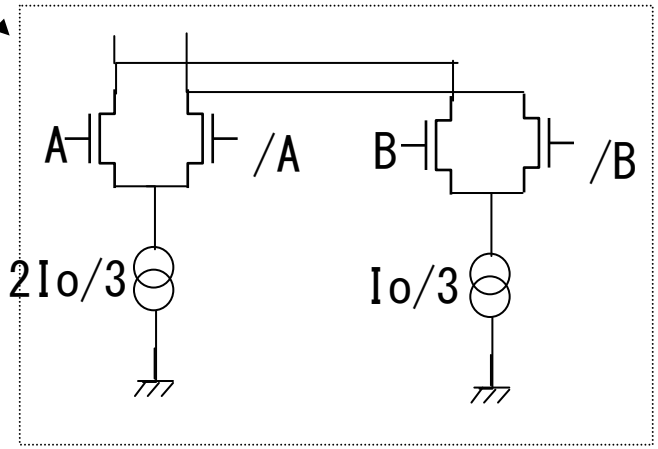
伝送路を駆動するDriverの回路例

CML (Current Mode Logic) 2値の例

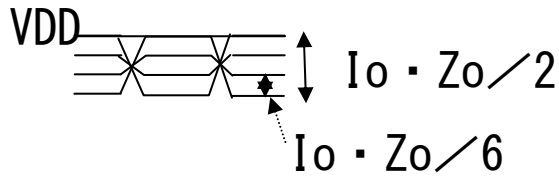


（ 差動振幅はこの倍の値 ）

CML 4値化の例



Receiver入力



（ 差動振幅はこの倍の値 ）

CML Driver の欠点利点

欠点：消費電力が非常に大きい

（次に示すNMOS Push-pull Driver比で、電流は2ないし4倍

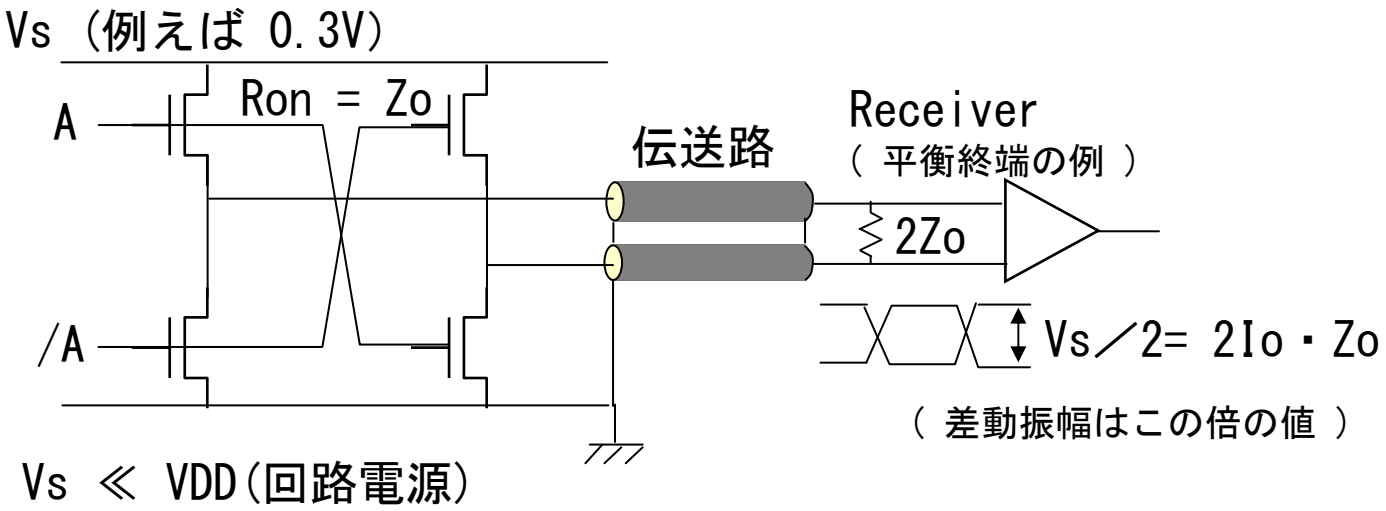
電圧も3～10倍、電力にして数十倍大きい ）

利点：多値化しても電源電流変化が無い

（Driverの電流変化は電源GNDのバウンスをもたらす）

NMOS Push-pull Driverの多値化

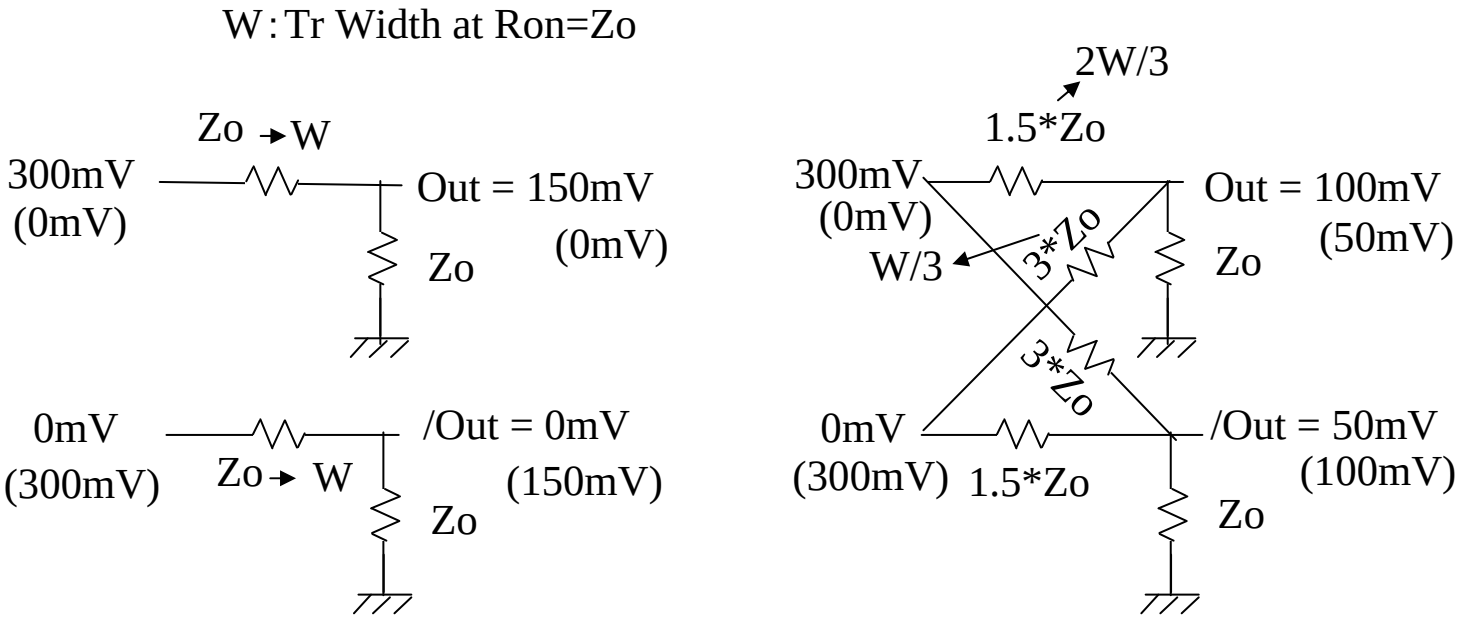
ベースとなる2値Driver

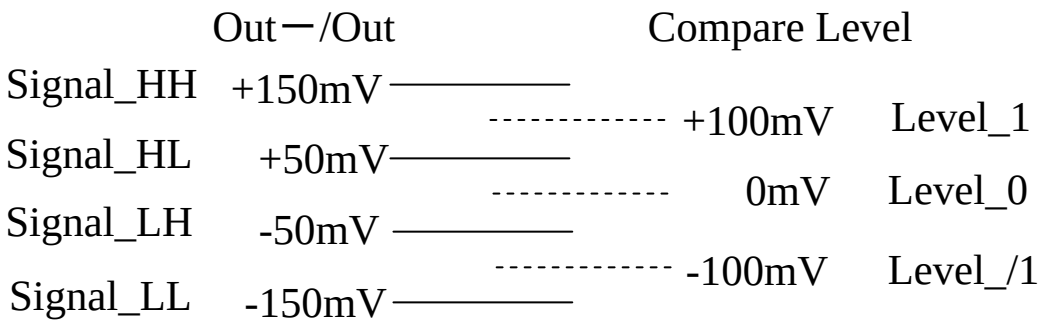


消費電流Io＝Vs／(4Zo)

（ 同じ振幅なら電流は1/4で済む ）

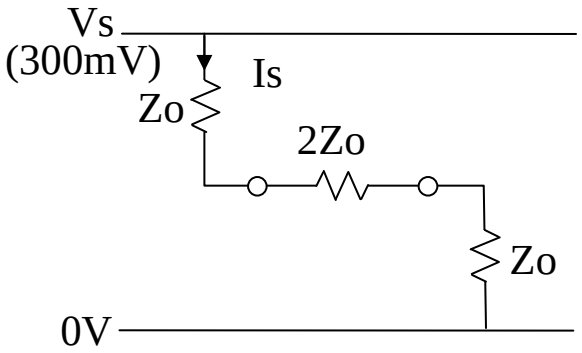
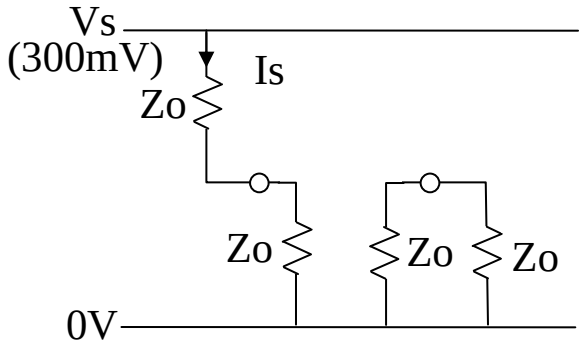
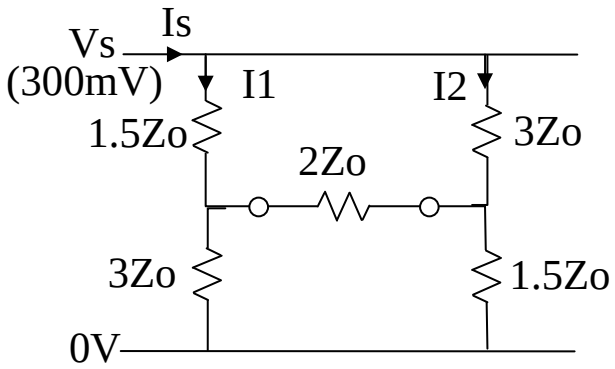
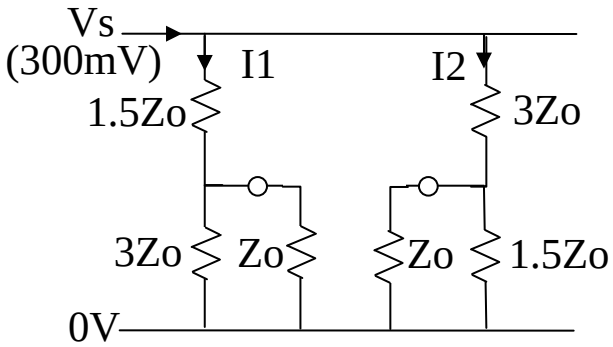
4値 Driver の原理





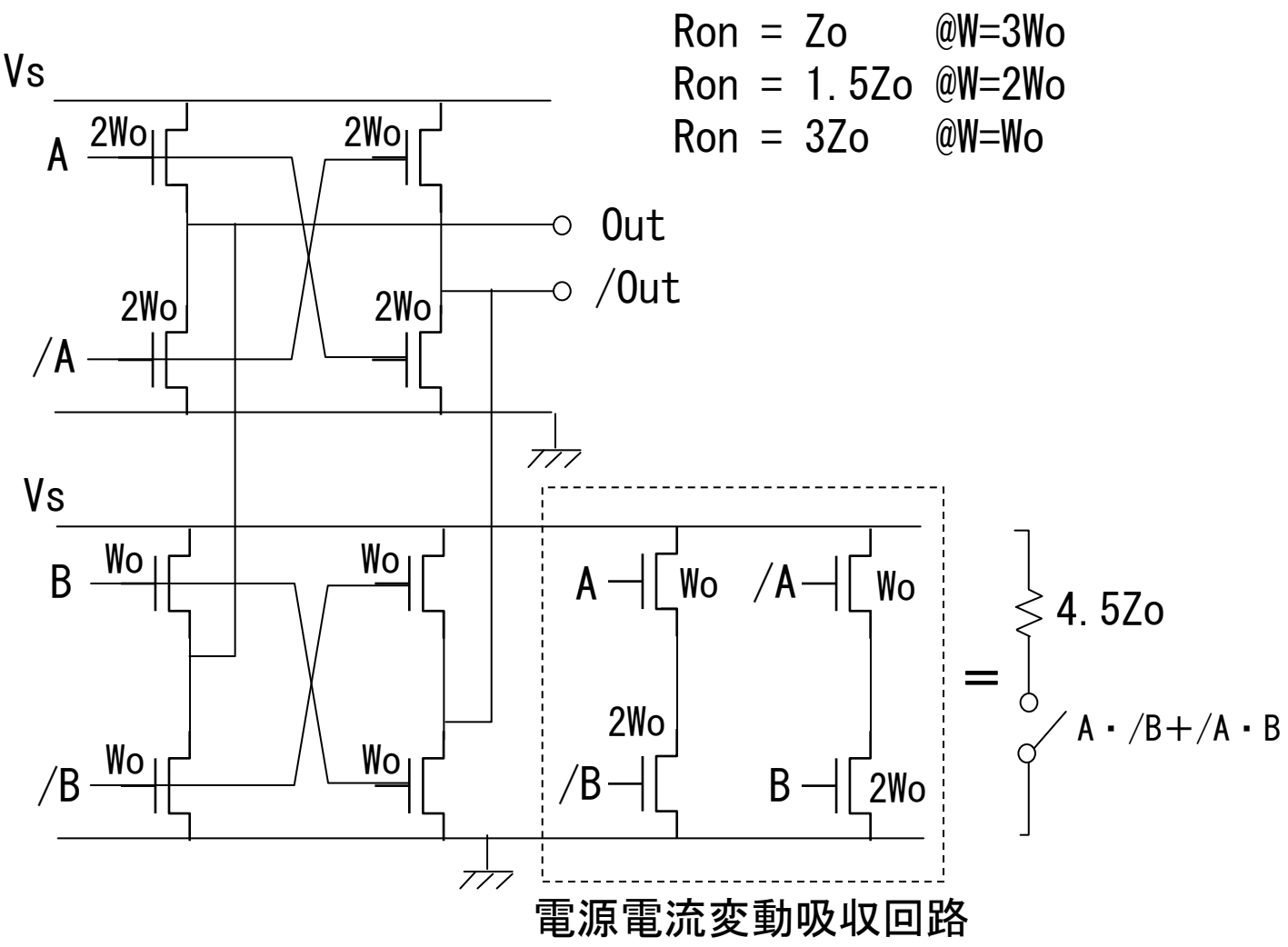
4値 Driver の Vs (0.3V)電源電流変動

Zo = 50 Ω

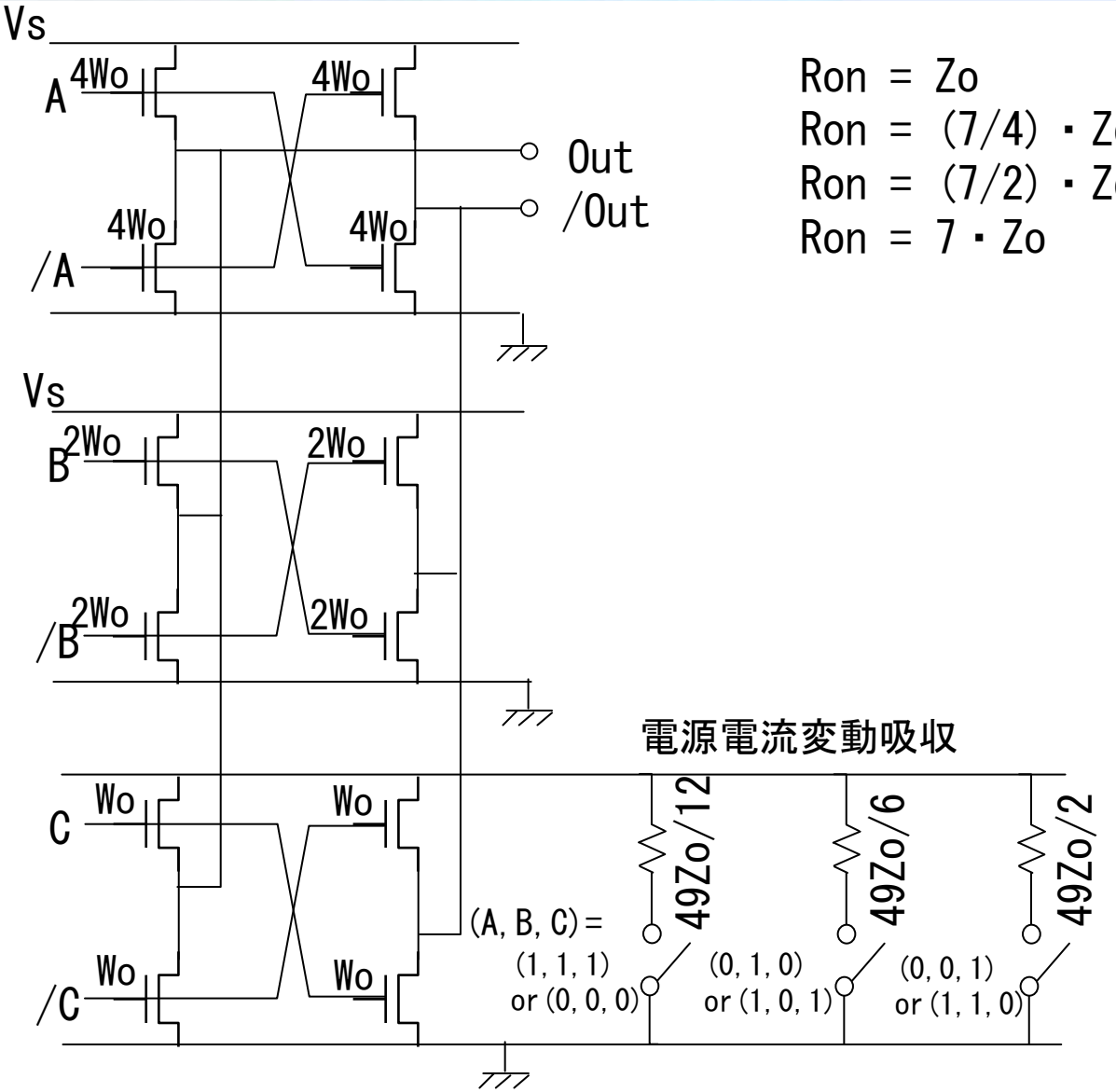
	平衡終端	独立終端
2値駆動状態	 Is = Vs ／4 Zo = 1.5 mA	 Is = Vs ／2 Zo = 3. mA
4値中間駆動状態	 I1 = (5Vs ／12)／1.5Zo = 1.666 mA I2 = (7Vs ／12)／3Zo = 1.166 mA Is = I1＋I2 = 2.833 mA = 1.5 mA × 1.89 = 1.5 mA＋1.333 mA	 I1 = Vs ／(1.5+0.75)／Zo = 2.666 mA I2 = Vs ／(3+0.6)／Zo = 1.666 mA Is = I1＋I2 = 4.333 mA = 3 mA × 1.44 = 3 mA＋1.333 mA

(1.333 mA = Vs／4.5Zo)

4値 Driver の 具体回路（電源電流変動吸収回路付き）

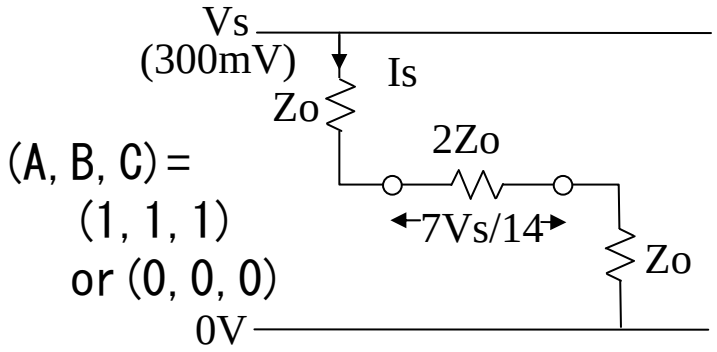


8値 Driver の 例（電源電流変動吸収回路付き）



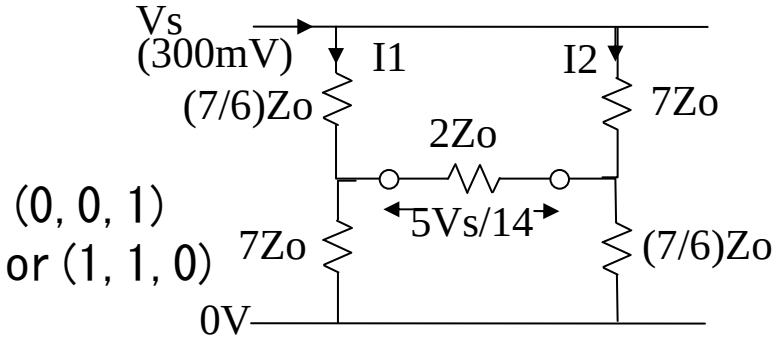
Ron = Zo @W=7Wo
Ron = (7/4)・Zo @W=4Wo
Ron = (7/2)・Zo @W=2Wo
Ron = 7・Zo @W=Wo

8値 Driver の Vs (0.3V)電源電流変動 (平衡終端)



Is = Vs ／4 Zo

Is + Vs ／(49Zo/12) = (97/49)*(Vs/4Zo)

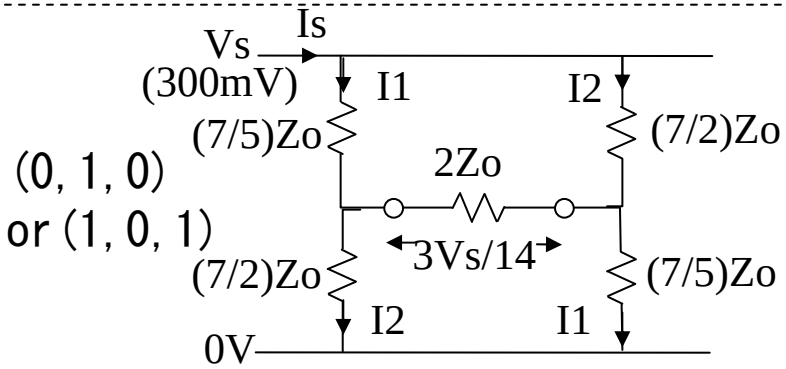


I1 = (4.5Vs /14)／(7Zo/6) = (54/49)*(Vs/4Zo)

I2 = (9.5Vs /14)／(7Zo) = (19/49)*(Vs/4Zo)

Is = I1＋I2 = (73/49)*(Vs/4Zo)

Is + Vs ／(49Zo/2) = (97/49)*(Vs/4Zo)

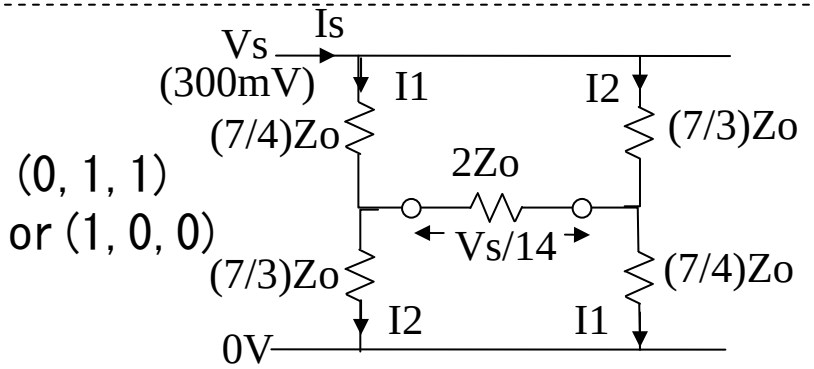


I1 = (5.5Vs /14)／(7Zo/5) = (55/49)*(Vs/4Zo)

I2 = (8.5Vs /14)／(7Zo/2) = (34/49)*(Vs/4Zo)

Is = I1＋I2 = (89/49)*(Vs/4Zo)

Is + Vs ／(49Zo/6) = (97/49)*(Vs/4Zo)



I1 = (6.5Vs /14)／(7Zo/4) = (52/49)*(Vs/4Zo)

I2 = (7.5Vs /14)／(7Zo/3) = (45/49)*(Vs/4Zo)

Is = I1＋I2 = (97/49)*(Vs/4Zo)