A19

逐次比較近似ADCにおける整数論を用いた冗長アルゴリズム設計

Redundancy Algorithm Design for an SAR ADC Using Number Theory

荒船拓也　小林佑太朗　澁谷将平　小林春夫

群馬大学大学院理工学府　電子情報部門

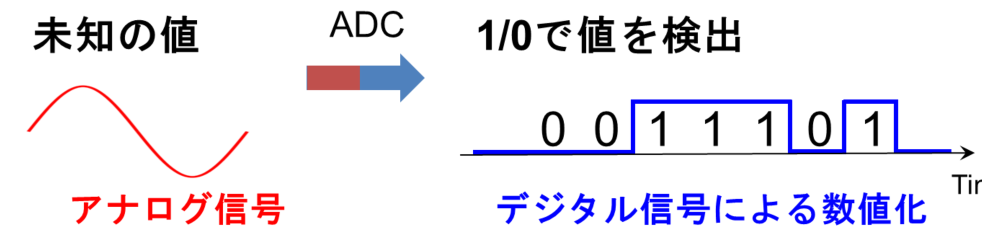
自動車応用

研究背景・目的

近年自動車のエレクトロニクス化が著しい

Problem

車載用途のアナログ・デジタル変換器（ADC）の要求が厳しい



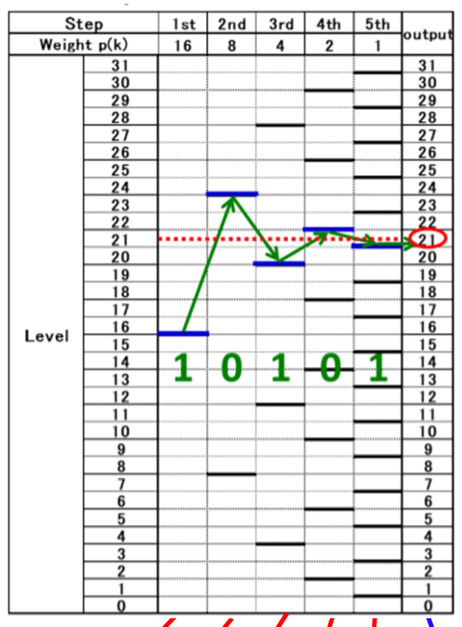
Approach
+整数論

Objective

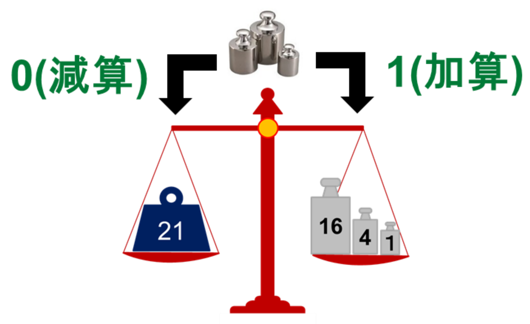
逐次比較近似AD変換器の**フィボナッチ数列**を用いた冗長設計

⇨ 高性能化・高速化

どうやって変換するの？



逐次比較＝天秤の原理



出力とディジタル表現
1対1

誤判定補正不可
（低信頼性）

必ず引く

16 + 8 - 4 + 2 - 1 + 0.5 - 0.5 = 21

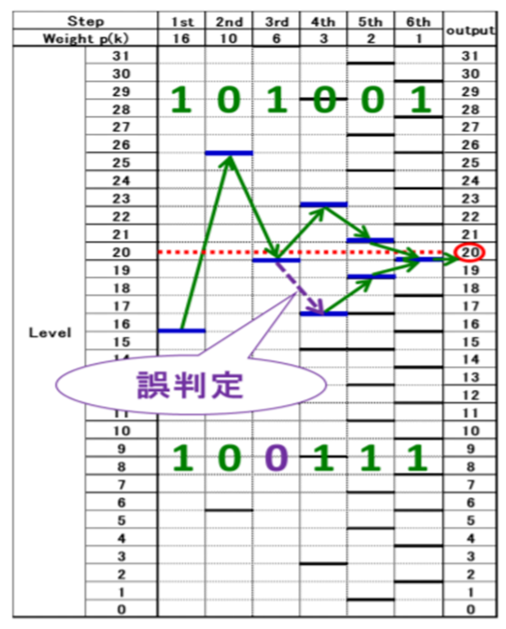
冗長性で補正力アップ!?

冗長＝余分・余裕

時間の冗長性の利用
- 比較回数を増加
- 電圧おもりの変更

出力とディジタル表現
1対複数

誤判定補正(高信頼性)
判定時間短縮(高速化)



20 = { 16 + 10 - 6 + 3 - 2 - 1 + 0.5 - 0.5
16 + 10 - 6 - 3 + 2 + 1 + 0.5 - 0.5 }

フィボナッチ数列

おもり決定の従来手法

基数選択法(Radix手法)

電圧おもり $p(k) = r^{M-k}$ (here $1 \le r < 2$)

radix 1.8	×1.8	×1.8	×1.8				
weight	34.0	18.9	10.5	5.8	3.2	1.8	1

整数丸め

$p(k) = 34 \quad 19 \quad 10 \quad 6 \quad 3 \quad 2 \quad 1$

小数 Radix { 誤判定の**補正可能**
おもりが必ず小数☹ }

フィボナッチ数列ってなに？

定義

$$F_{n+2} = F_n + F_{n+1}$$

[ただし、$F_0 = 0,\ F_1 = 1$, n=0,1,2...]

Leonardo Fibonacci
(伊:1170~1250年頃)

性質

隣接2項の比率は**黄金比**(1.618…)に収束

ex. 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55 ... (×1.62, ×1.62)

Point

整数のみで**小数Radix**を実現!!

数学を回路に応用!?

提案手法

フィボナッチ数列を電圧おもりへ応用

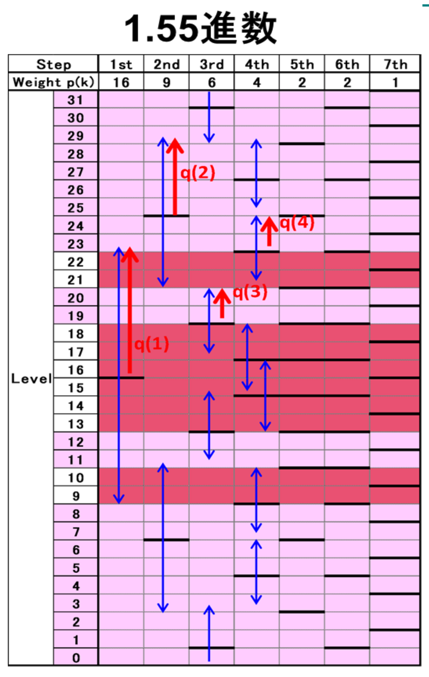
radix 1.62	×1.62	×1.62	×1.62				
Fibonacci weight	13	8	5	3	2	1	1

フィボナッチ数列 { 誤判定の**補正可能**
おもりが**整数**☺ }

誤判定の補正

冗長性による判定誤り補正

無駄あり
1.55進数



補正力の定義

kステップ目の補正可能範囲: $q(k)$
- 入力電圧と比較電圧の最大差

補正可能条件

$|V_{in} - V_{com}| \le q(k)$ (V_{in} :入力電圧, V_{com} :入力電圧)

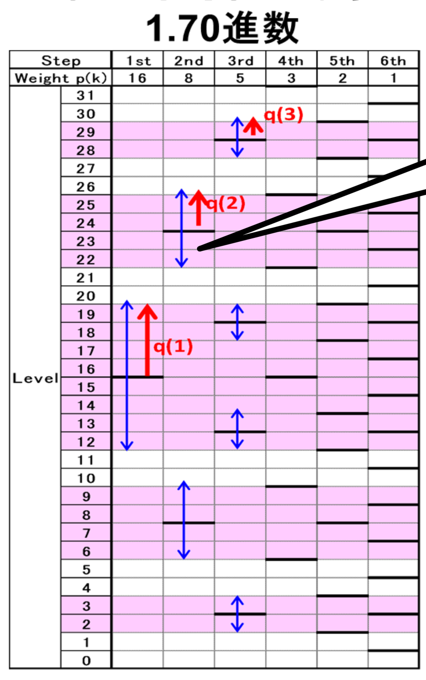
補正範囲の**無駄**が存在

補正不可能な入力範囲が存在

冗長設計効果の劣化

効率の悪い設計

補正不可範囲あり
1.70進数



フィボナッチ冗長設計の高信頼性

導出した補正式

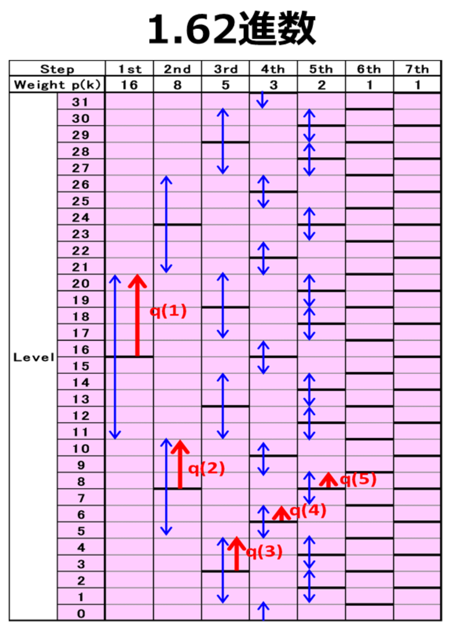
$$q(k) = -p(k+1) + 1 + \sum_{i=k+2}^{M} p(i)$$

(p(k):k step目の比較重み
M:総ステップ数)

新性質の発見

① 許容値は必ずフィボナッチ数
② 許容できる範囲が必ず接する
③ 1.62は冗長度の境界条件
- Radix < 1.62 ⇒ 無駄な補正範囲発生
- Radix = 1.62 ⇒ **無駄なく全入力範囲をカバー**
- Radix > 1.62 ⇒ 補正できない入力有り

フィボナッチ冗長設計
1.62進数

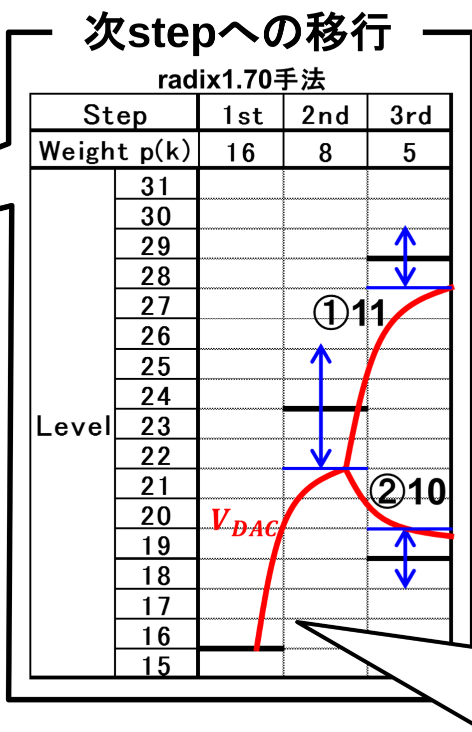


最も効率の良い冗長設計

判定までの整定時間

天秤回路の整定時間の定義

次stepへの移行



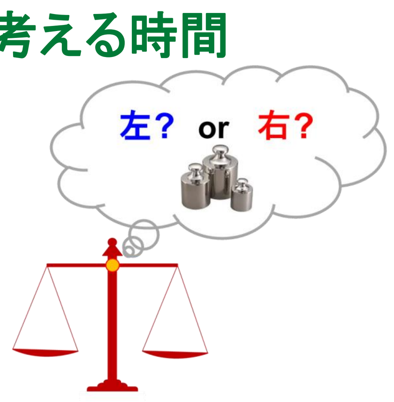
整定時間＝天秤が考える時間

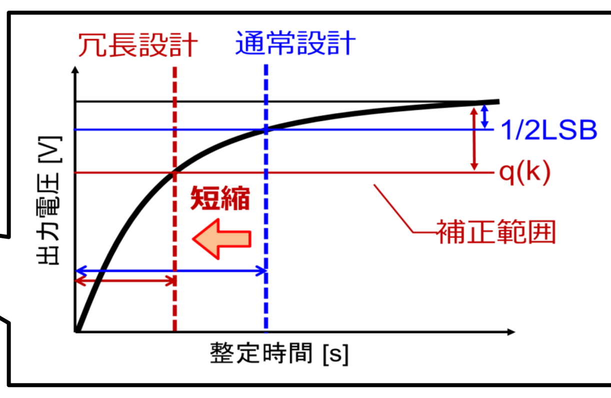
整定時間の定義

整定の種類

反転整定 or **非反転整定**

非反転時の整定時間を検証





一般の変換器
⇨ 整定時間：長

冗長設計の変換器
⇨ 整定時間：短

フィボナッチ手法による高速化

一般整定時間

$$T_{settle}(k) = \tau \ln\left(\frac{p(k)+q(k-1)}{q(k)}\right)$$

フィボナッチ重みを利用
$p(k) = F_{M-k+1}$
$q(k) = F_{M-k-1}$

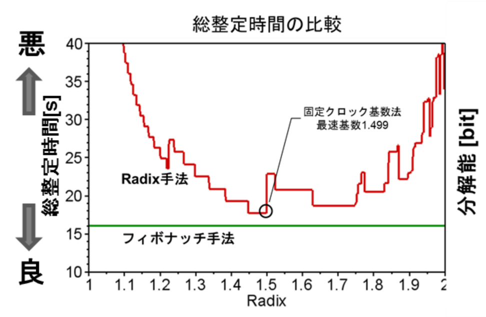
$$= \tau \ln\left(\frac{F_{M-k+1}+F_{M-k}}{F_{M-k-1}}\right)$$

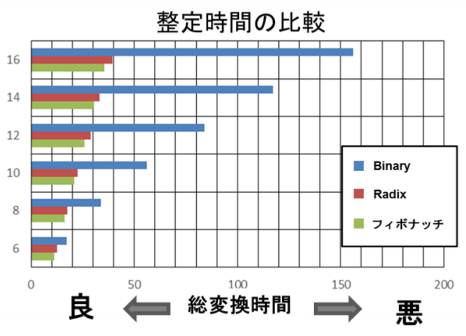
$$= \tau \ln\left(\frac{(F_{M-k}+F_{M-k-1})+F_{M-k}}{F_{M-k-1}}\right)$$

$$= \tau \ln\left(2\frac{F_{M-k}}{F_{M-k-1}}+1\right)$$

隣接項は黄金比
$\lim_{n\to\infty} \frac{F_n}{F_{n-1}} = 1.618... = \varphi$

$$\tau \ln(2\varphi + 1) = 1.444\tau$$





従来手法と比較しての変換時間が**短縮**
(Radix手法より10%短縮)

新性質の発見

$T_{settle}(k) = \tau \ln(2\varphi + 1) = 1.444\tau$

整定時間は常に一定

変換時間が最も短い

結論

◆フィボナッチ数列の冗長設計の有効性

- 全入力値の補正による**高信頼性**
- 広補正範囲による**高速変換**
- 1.62進数は冗長設計の**境界条件**

➡ **冗長設計に大きな貢献!!**

参考文献

[1].Y. Kobayashi, S. Shibuya, T. Arafune, S. Sasaki, H. Kobayashi : “SAR ADC Design Using Golden Ratio Weight Algorithm”, The 15th International Symposium on Communications and Information Technologies 2015, Nara, Japan (Oct. 2015)

[2].T. Arafune, , Y. Kobayashi, S. Shibuya, H. Kobayashi : “Fibonacci Sequence Weighted SAR ADC Algorithm and its DAC Topology”, The 11th IEEE International Conference on ASIC 2015, Chengdu, China, (Nov. 2015)