15th International Symposium
on Communications and Information Technologies

Oct. 09, 2015

SAR ADC Design Using Golden Ratio Weight Algorithm

Yutaro Kobayashi **Shohei Shibuya**
Takuya Arafune **Shu Sasaki** **Haruo Kobayashi**

Gunma University

Supported by STARC

Kobayashi Laboratory

群馬大学
GUNMA UNIVERSITY

Research Objective

Objective

- **Development of**
  Reliable & high-speed SAR ADC

Our Approach

- **Redundant search algorithm design with Number Theory**

Outline

- Research Background
- SAR ADC Redundancy Design
- Proposed SAR Algorithm Using Fibonacci Sequence
- Advantages of Proposed SAR Algorithm
- Realization of Fibonacci DAC
- Conclusions

Outline

- **Research Background**
- SAR ADC Redundancy Design
- Proposed SAR Algorithm Using Fibonacci Sequence
- Advantages of Proposed SAR Algorithm
- Realization of Fibonacci DAC
- Conclusions

Research Background

Automotive Electronics are in spotlight

High-speed, Reliable

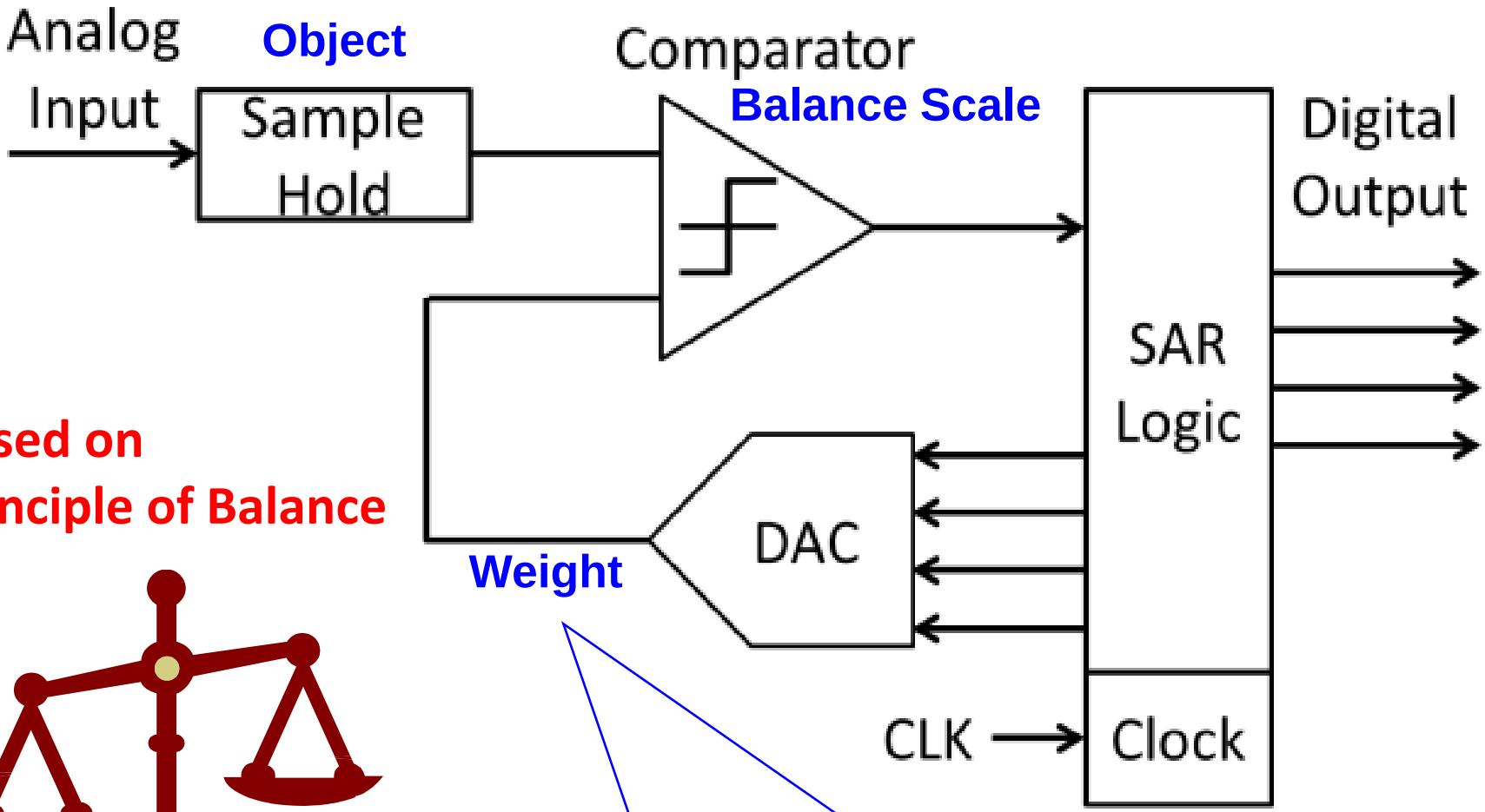
“SAR ADC” in microcontroller is needed

Redundancy design for error correction

Design issues

SAR ADC : Successive Approximation Register type Analog to Digital Converter

SAR ADC Configuration



Based on Principle of Balance

Generally use binary weight

(1 , 2 , 4 , 8 , 16 , 32, 64 …)

Binary Search SAR ADC

5bit-5step SAR ADC

Input voltage Vin (Object) 7.3

Reference voltage weight p(k)

Weight **p(1)** 16

Weight **p(2)** 8

Weight **p(3)** 4

Weight **p(4)** 2

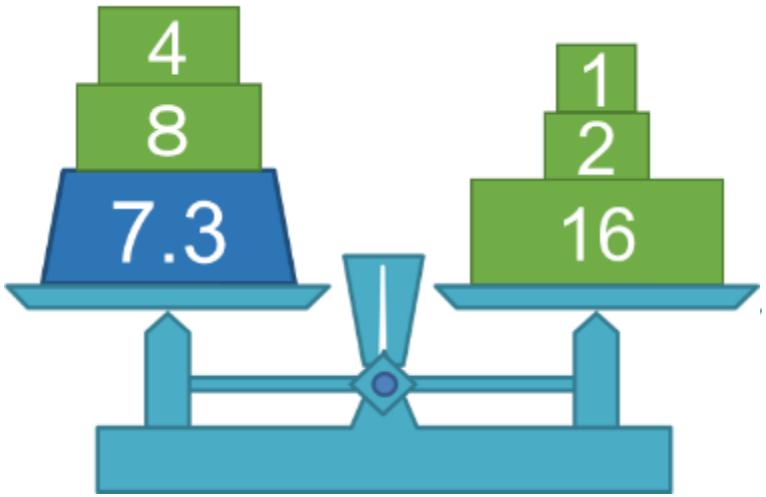
Weight **p(5)** 1

Step		1st	2nd	3rd	4th	5th	output
Weight p(k)		16	8	4	2	1	
Level	31						31
	30						30
	29						29
	28						28
	27						27
	26						26
	25						25
	24						24
	23						23
	22						22
	21						21
	20						20
	19						19
	18						18
	17						17
	16						16
	15						15
	14						14
	13						13
	12						12
	11						11
	10						10
	9						9
	8						8
	7						7
	6						6
	5						5
	4						4
	3						3
	2						2
	1						1
	0						0

Binary Search SAR ADC Operation(1)

5bit-5step SAR ADC

- **Binary weight**
  p(k) = 16, 8, 4, 2, 1
- **Analog input 7.3**



7.3 ⇒ 7

Step	1st	2nd	3rd	4th	5th	output
Weight p(k)	16	8	4	2	1	
Level 31						31
30						30
29						29
28						28
27						27
26						26
25						25
24						24
23						23
22						22
21						21
20						20
19						19
18						18
17						17
16						16
15						15
14						14
13						13
12	−8					12
11						11
10						10
9						9
8						8
7						7
6					+1	6
5		−4		+2		5
4						4
3						3
2	0	0	1	1	1	2
1						1
0						0

Binary Search SAR ADC Operation(2)

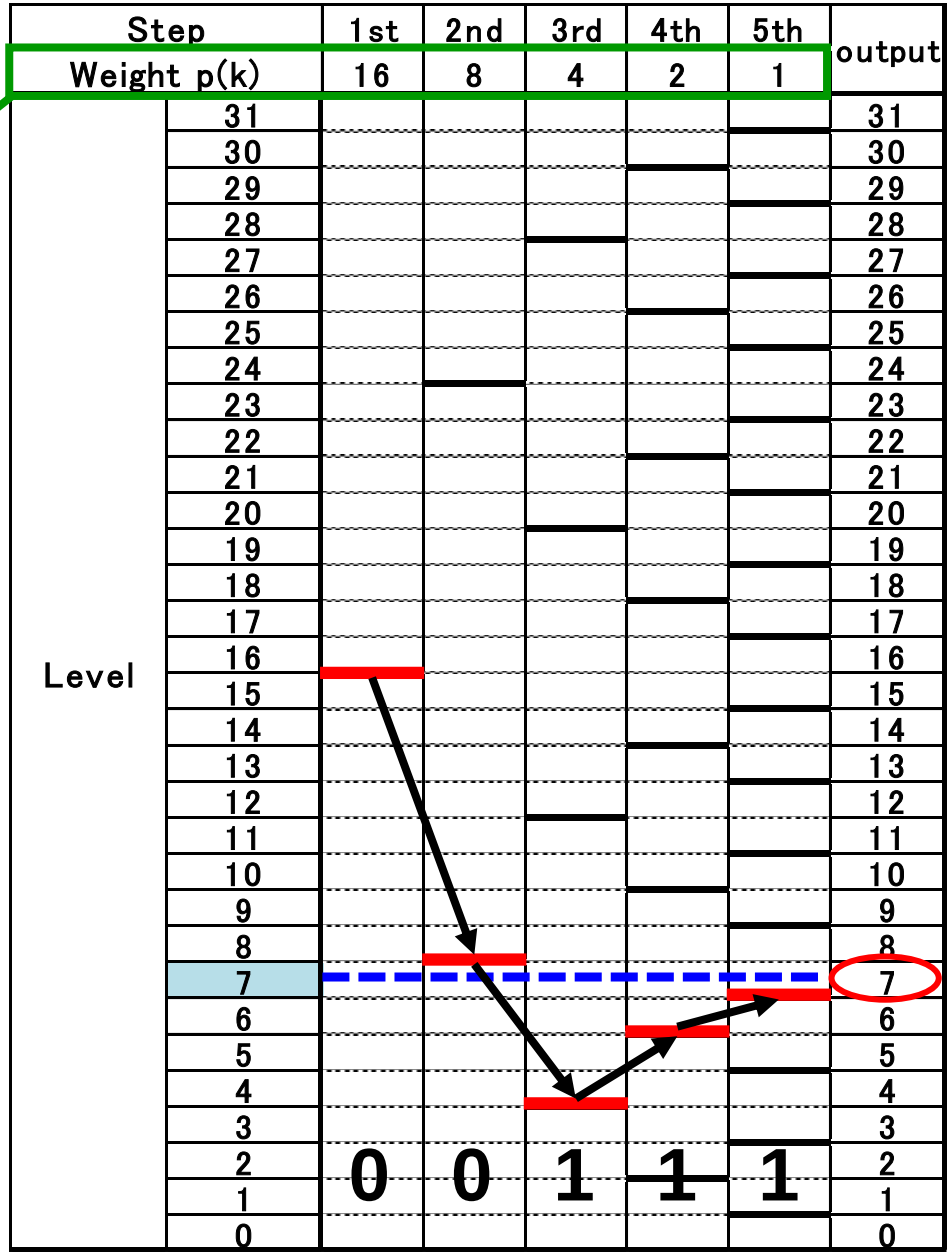
5bit-5step SAR ADC

- **Binary weight**
  p(k) = 16, 8, 4, 2, 1
- **Analog input 7.3**

One-to-one mapping
between decimal and binary codes

$$D_{out} = (00111)_2$$

$$7 = 16 - 8 - 4 + 2 + 1 + 0.5 - 0.5$$



Binary Search SAR ADC Operation(3)

5bit-5step SAR ADC

- Binary weight
  p(k) = 16, 8, 4, 2, 1
- Analog input 7.3

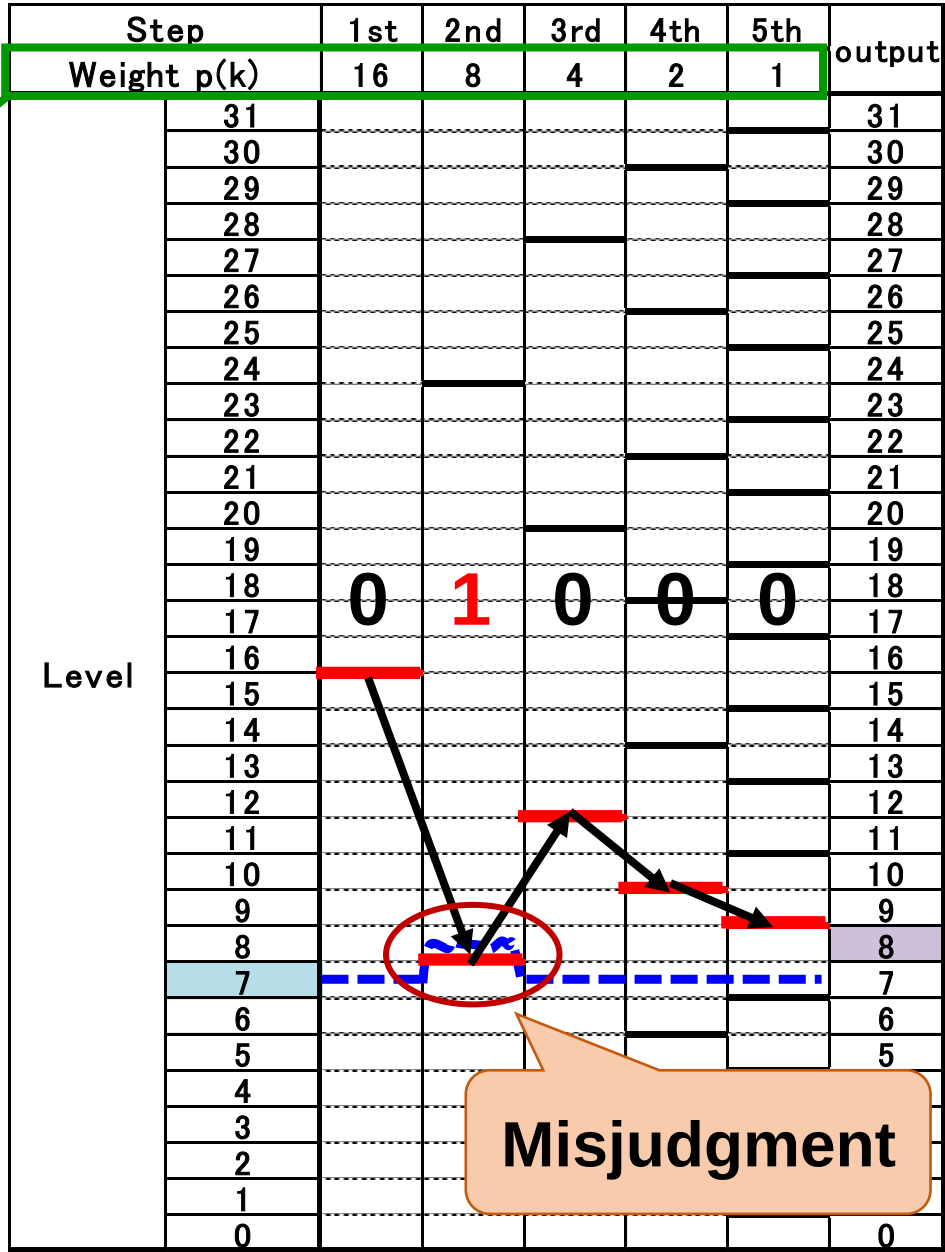
One-to-one mapping
between decimal and binary codes

D_{out} =(00111)$_2$=7

1 misjudgment leads to
incorrect output

D_{out} =(01000)$_2$=8

Reliability problem !



Outline

- Research Background
- **SAR ADC Redundancy Design**
- Proposed SAR Algorithm Using Fibonacci Sequence
- Advantages of Proposed SAR Algorithm
- Realization of Fibonacci DAC
- Conclusions

SAR ADC Redundancy Design

Redundancy: Surplus, Extra

Apply to SAR ADC

Using time redundancy

- **Increase comparison steps**
- **Change reference voltages**

Multiple output expressions

Binary weight
p(k): 1,2,4,8,16

Non-binary weight
p(k): 1,2,3,6,10,16

Enable digital error correction!

Redundant Search SAR ADC Operation(1)

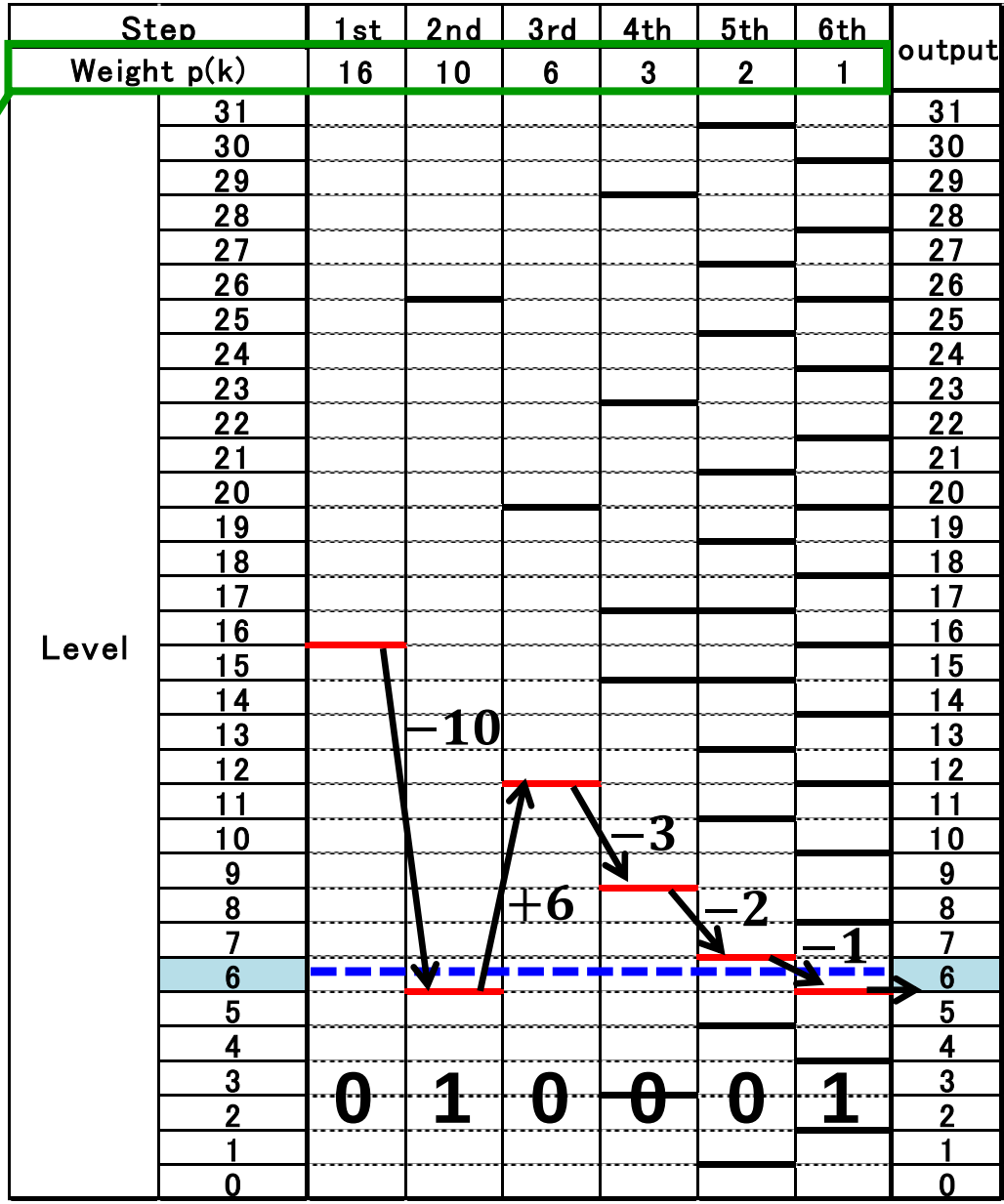
5bit-6step SAR ADC

- **Redundant weight**
  p(k) = 16, 10, 6, 3, 2, 1
- **Analog input 6.3**

Increase number of comparison steps

6⇒010001⇒6

16−10+6−3−2−1+0.5-0.5
= 6



Redundant Search SAR ADC Operation(2)

5bit-6step SAR ADC

- Redundant weight
  p(k) = 16, 10, 6, 3, 2, 1
- Analog input 6.3

Increase number of comparison steps

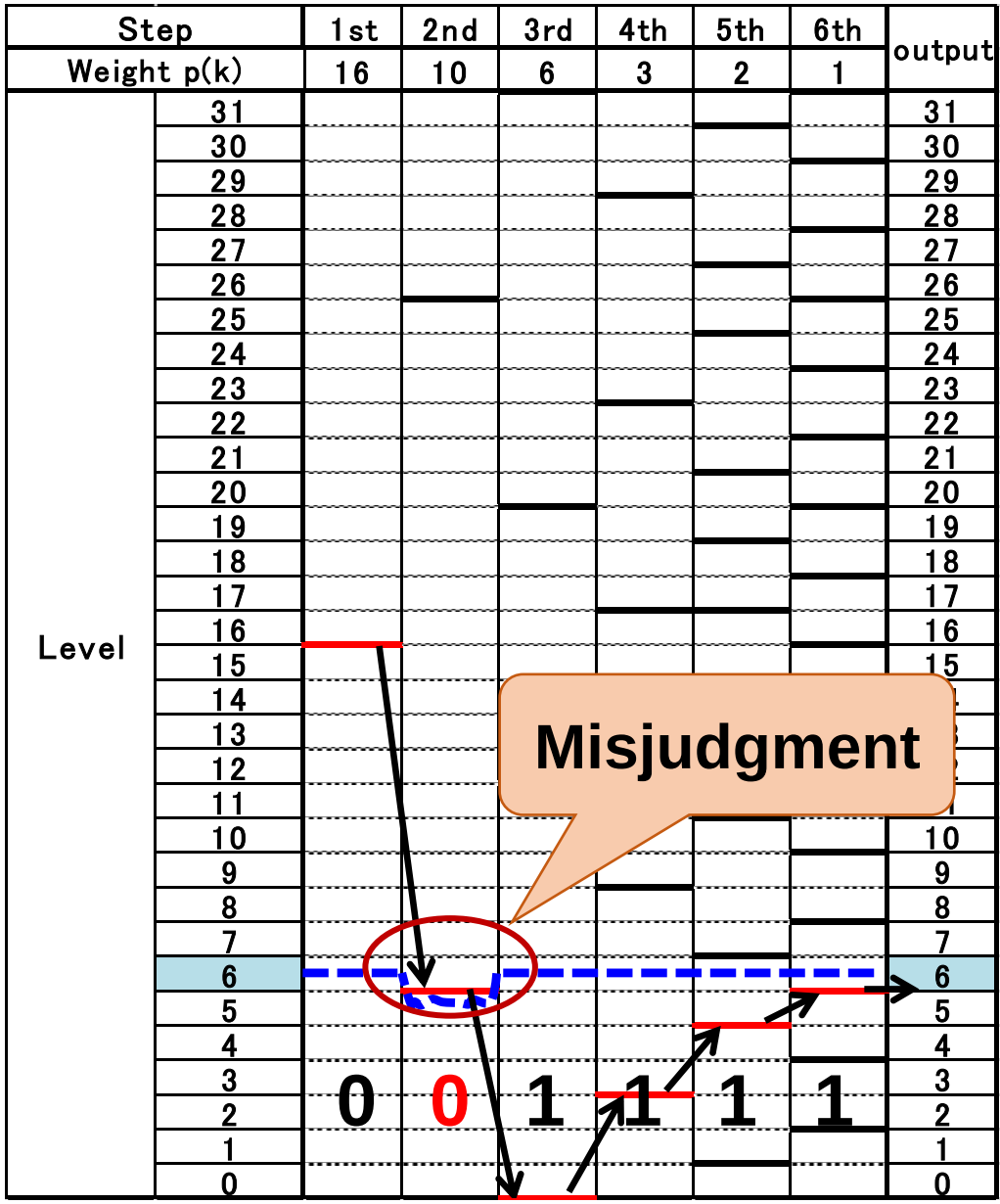
6⇒010001⇒6

Another expression

6⇒001111⇒6

Error correction

⇒ High-Reliability



Evaluation of Redundancy Design

5bit-6step SAR ADC

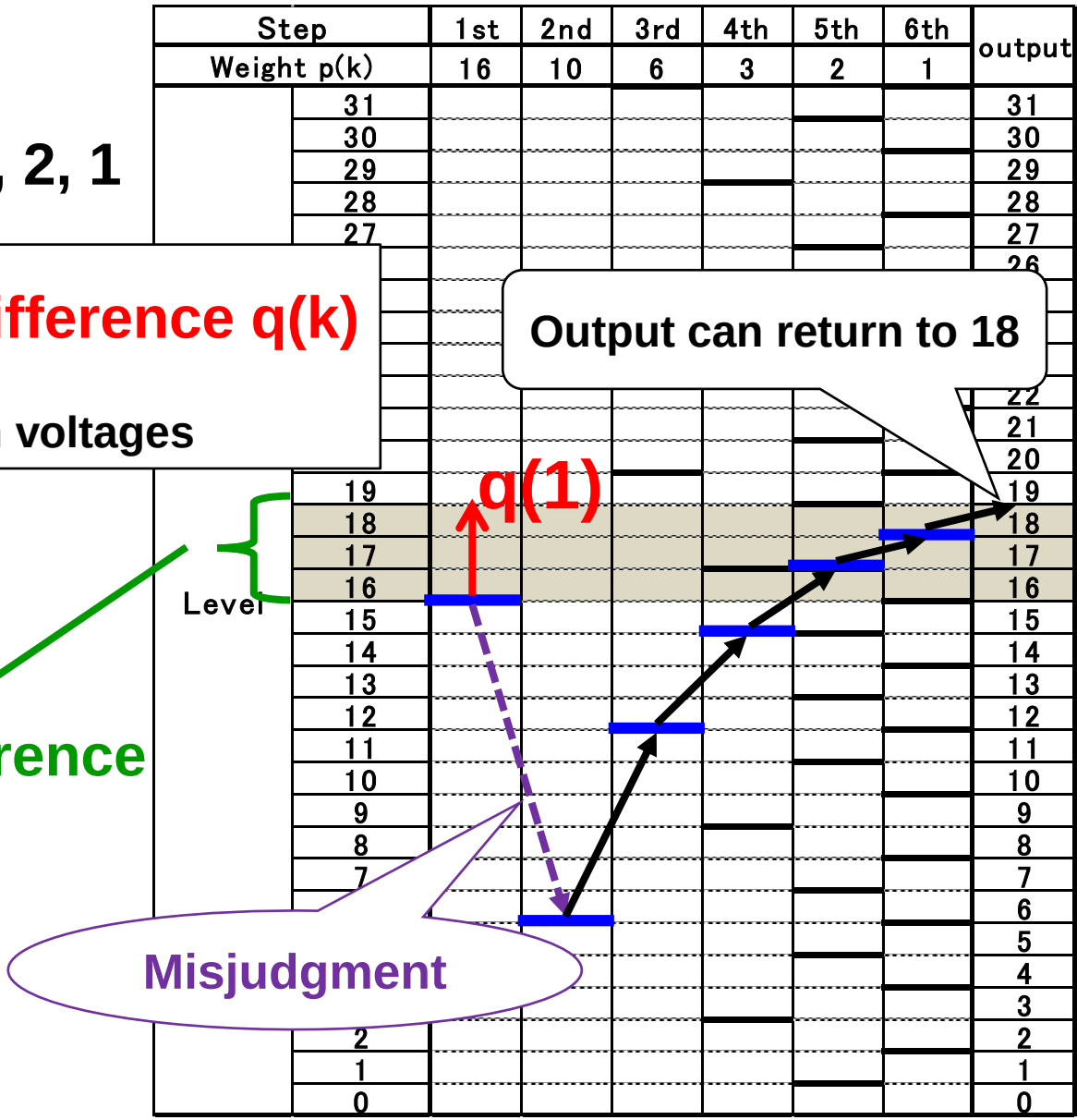
- **Redundant weight**
  p(k) = 16, 10, 6, 3, 2, 1

Definition

k-th step correctable difference q(k)

q(k) is correctable difference between input and comparison voltages

1st step correctable difference
q(1)=3



Correctable Difference q(k)

5bit-6step SAR ADC

- **Redundant weight**
  p(k) = 16, 10, 6, 3, 2, 1

Definition

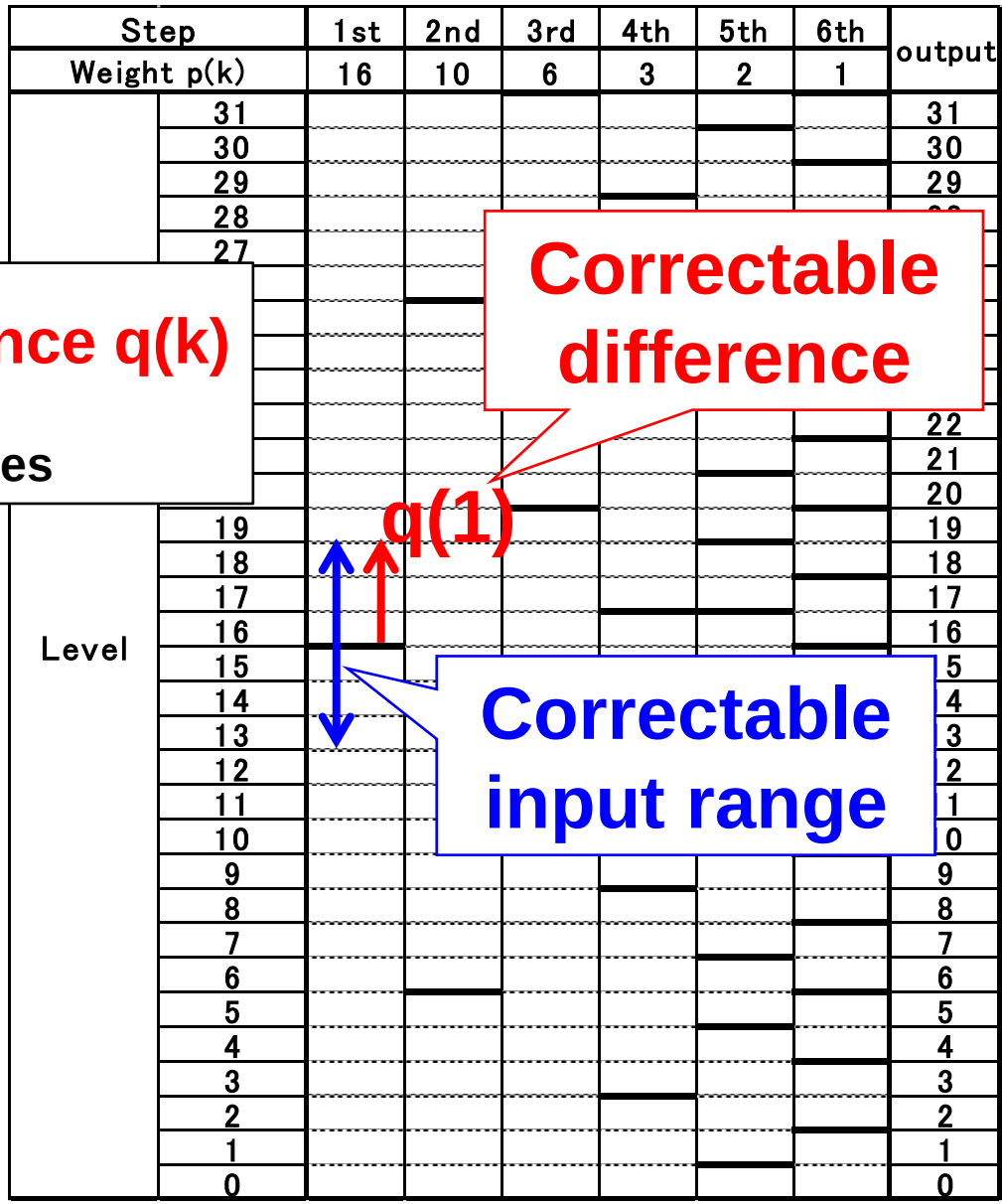
k-th step correctable difference q(k)

q(k) is correctable difference between input and comparison voltages

Condition of correctable

$$|Vin - Vcom(k)| \leq q(k)$$

Large q(k) indicates High-Reliability



Correctable Difference q(k)

Formula for q(k)

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

p(k) : reference voltage weight at k-th step
M : number of total steps

q(k) determined only by p(k) !

Select proper p(k)

Step	1st	2nd	3rd	4th	5th	6th	output
Weight p(k)	16	10	6	3	2	1	
31							31
30							30
29							29
28							28
27							27
26							26
25							25
24							24
23							23
22							22
21							21
20							20
19							19
18							18
17							17
16							16
15							15
14							14
13							13
12							12
11							11
10							10
9							9
8							8
7							7
6							6
5							5
4							4
3							3
2							2
1							1
0							0

Level

q(1)

q(2)

q(3)

Conventional Method of p(k) Selection

We select N-bit and M-step SAR ADC k-th step reference voltage p(k).
here $p(1) = 2^{N-1}$

Conventional method

Radix method

$$p(k) = r^{M-k} \text{ (here } 1 \leq r < 2\text{)}$$

Problems

- Difficult to select good radix
  - Trade-off of conversion steps and correction capability
- p(k) must be fraction
  - Rounding to integer causes dispersion of q(k)

Issues of Conventional Method

5bit-6step SAR ADC

- **Radix method**

Radix=1.8

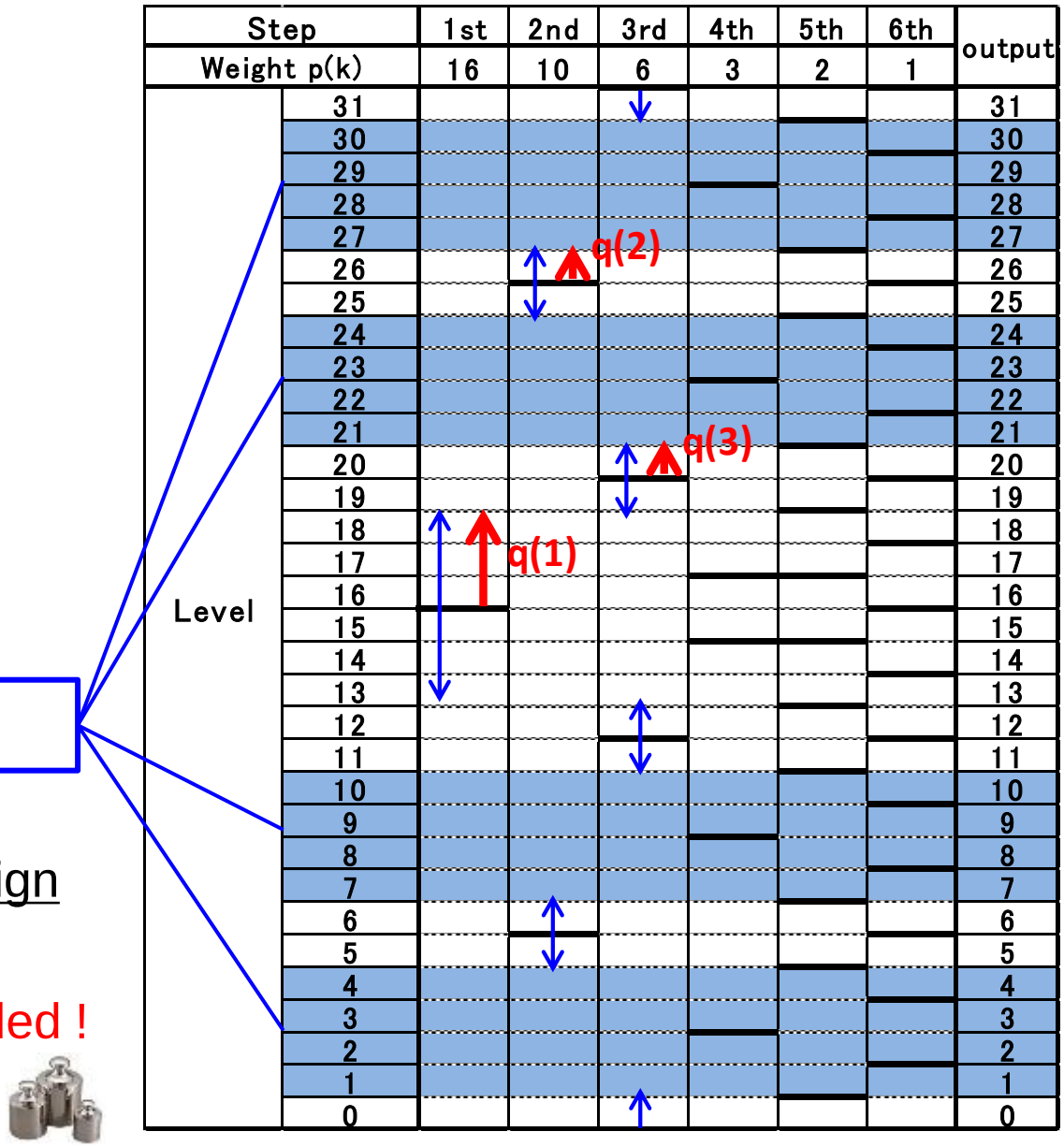
Reference voltage weight p(k)

$$p(1) = 2^{5-1} = 16$$
$$p(2) = 1.8^{4} \cong 10$$
$$p(3) = 1.8^{3} \cong 6$$
$$p(4) = 1.8^{2} \cong 3$$
$$p(5) = 1.8^{1} \cong 2$$
$$p(6) = 1.8^{0} = 1$$

uncorrectable range

Not effective redundancy design

Good selection method is needed !



Outline

- Research Background
- SAR ADC Redundancy Design
- **Proposed SAR Algorithm Using Fibonacci Sequence**
- Advantages of Proposed SAR Algorithm
- Realization of Fibonacci DAC
- Conclusions

Fibonacci Sequence

Definition (n=0,1,2,3…)

$$F_0 = 0$$
$$F_1 = 1$$
$$F_{n+2} = F_n + F_{n+1}$$

Example of numbers(Fibonacci number)

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89…

Leonardo Fibonacci (around 1170-1250)

Property

The closest terms ratio converges to *"Golden Ratio"* !

$$\lim_{n\to\infty} \frac{F_n}{F_{n-1}} = 1.618033988749895 = \varphi$$

Proposed Method of p(k) Selection

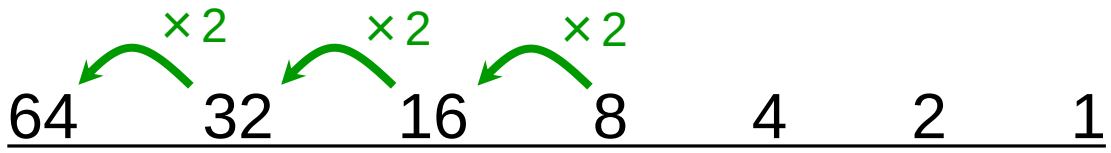
We select N bit and M step SAR ADC k-th step reference voltage p(k).

here $p(1) = 2^{N-1}$

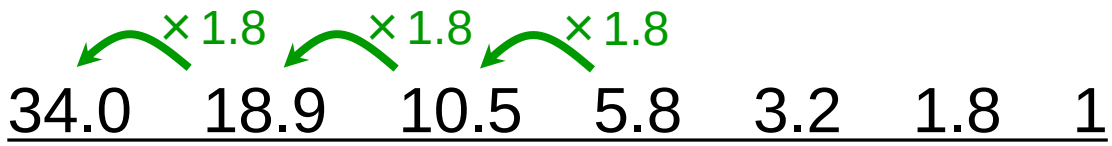
Proposed solution

Using Fibonacci sequence for p(k):$p(k) = F_{M-k+1}$

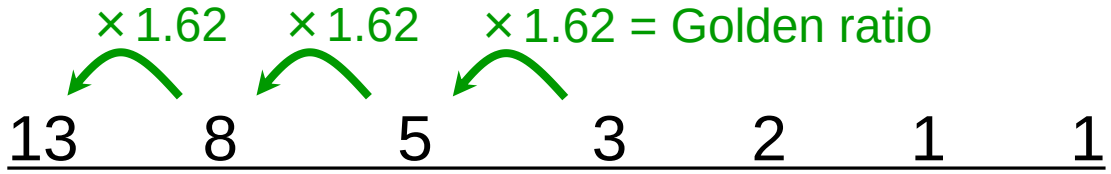
Binary Weight



Radix 1.8 Weight



Fibonacci Weight
(Radix 1.62 Weight)



Property converging to Golden Ratio

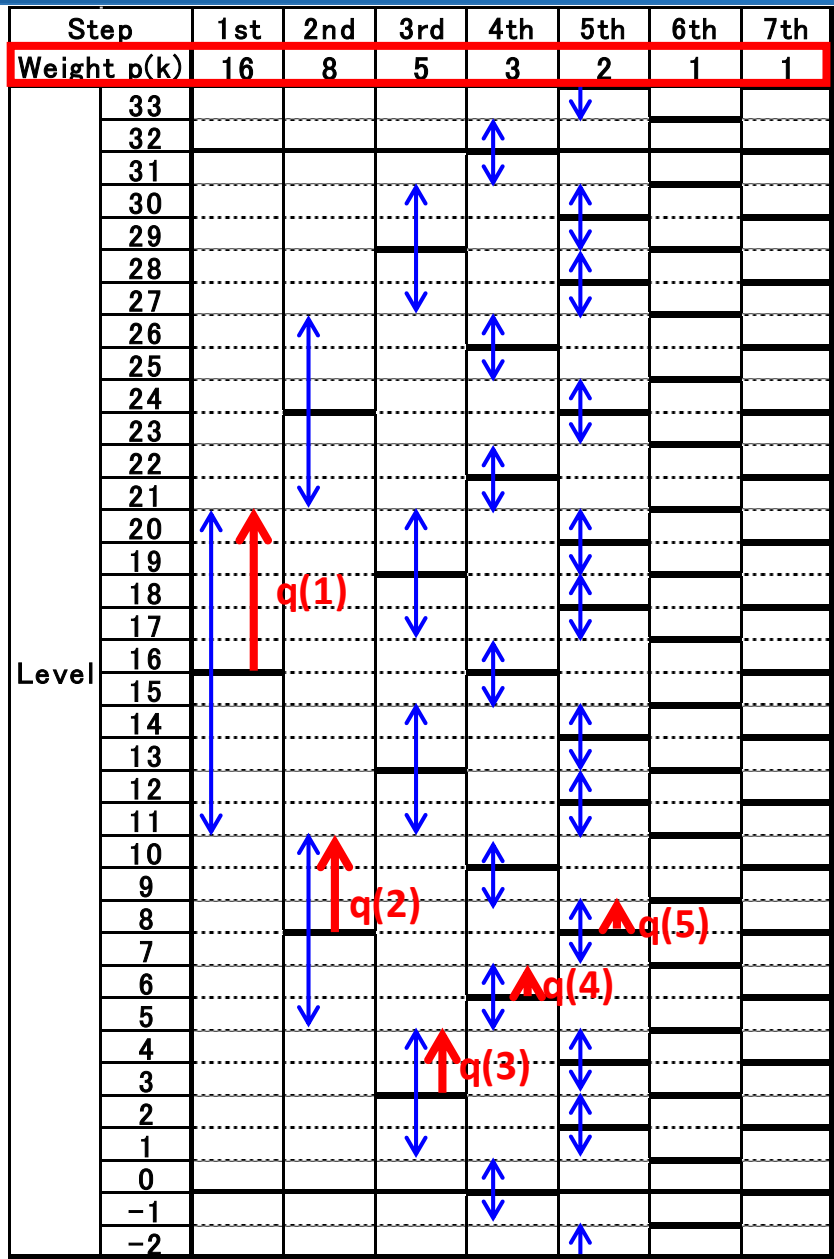
➡ Realize **Radix 1.62 Weight** by using **only integer** !

SAR ADC Using Fibonacci Sequence

Fibonacci sequence SAR ADC

Two properties are discovered !!

1. ***Correctable difference q(k) is always Fibonacci number*** F_{M-k-1}.
2. ***q(k) is exactly in contact q(k+1) without overlap.***



Properties of Fibonacci Sequence Usage

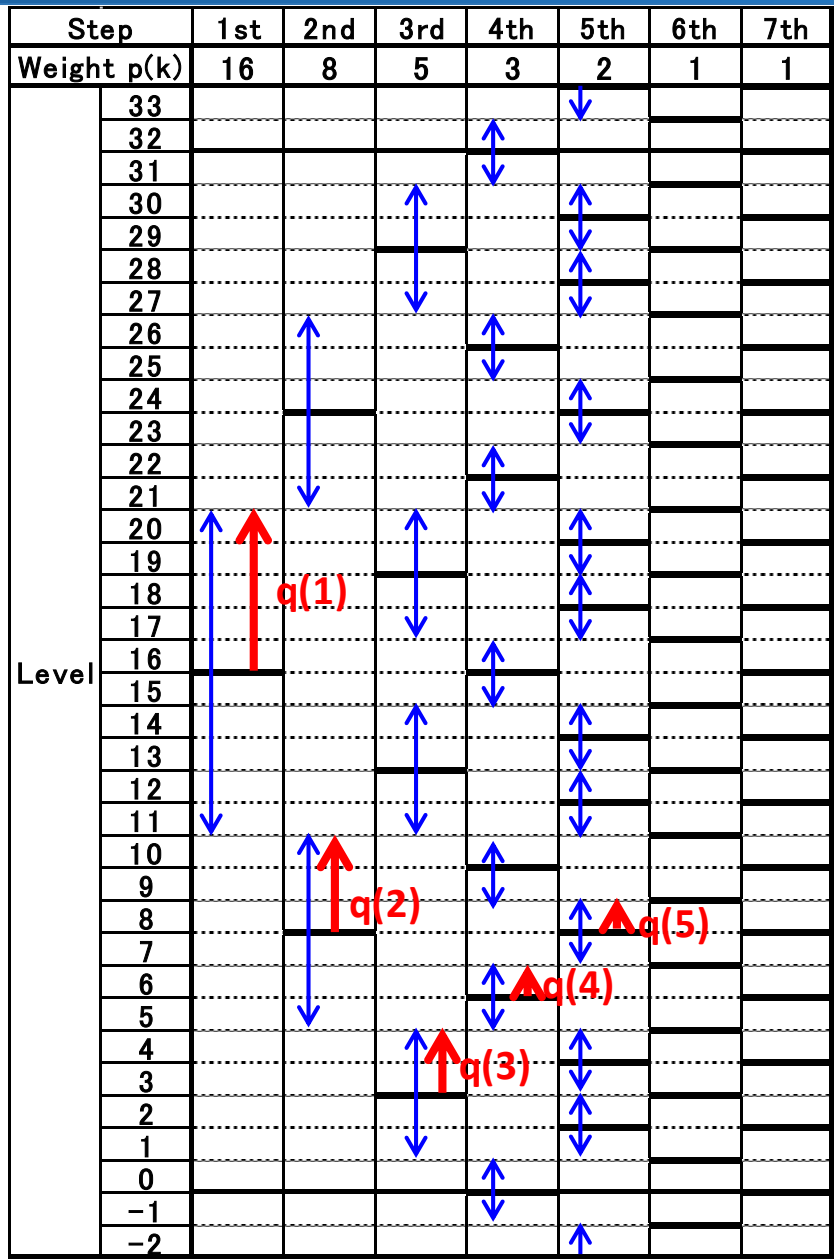
Fibonacci sequence SAR ADC

Two properties are discovered !!

1. ***Correctable difference q(k) is always Fibonacci number*** F_{M-k-1}.
2. *q(k) is exactly in contact q(k+1) without overlap.*

Correctable difference q(k)

$$q(1) = 5 = F_5$$
$$q(2) = 3 = F_4$$
$$q(3) = 2 = F_3$$
$$q(4) = 1 = F_2$$
$$q(5) = 1 = F_1$$



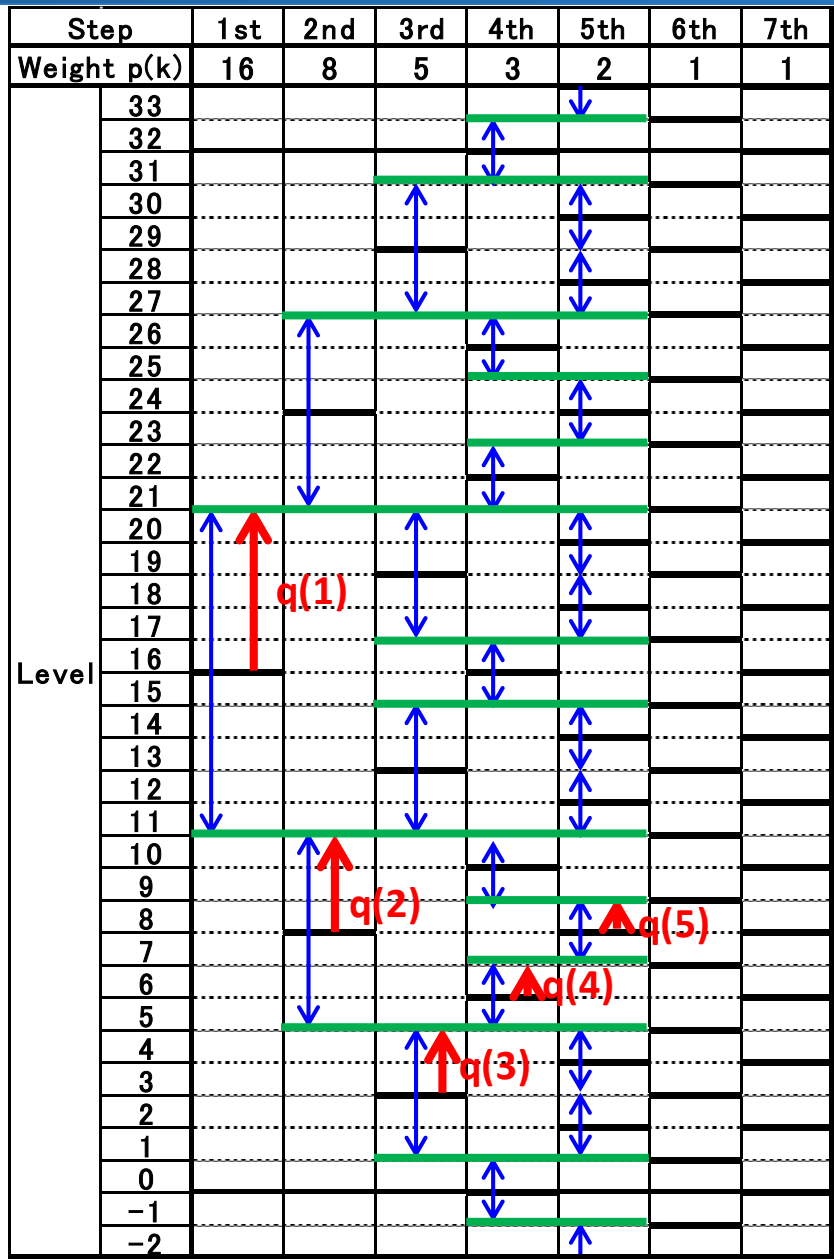
Properties of Fibonacci Sequence Usage

Fibonacci sequence SAR ADC

Two properties are discovered !!

1. *Correctable difference q(k) is always Fibonacci number* F_{M-k-1}.

2. ***q(k) is exactly in contact q(k+1) without overlap.***



Significance of Property 2

Fibonacci sequence SAR ADC

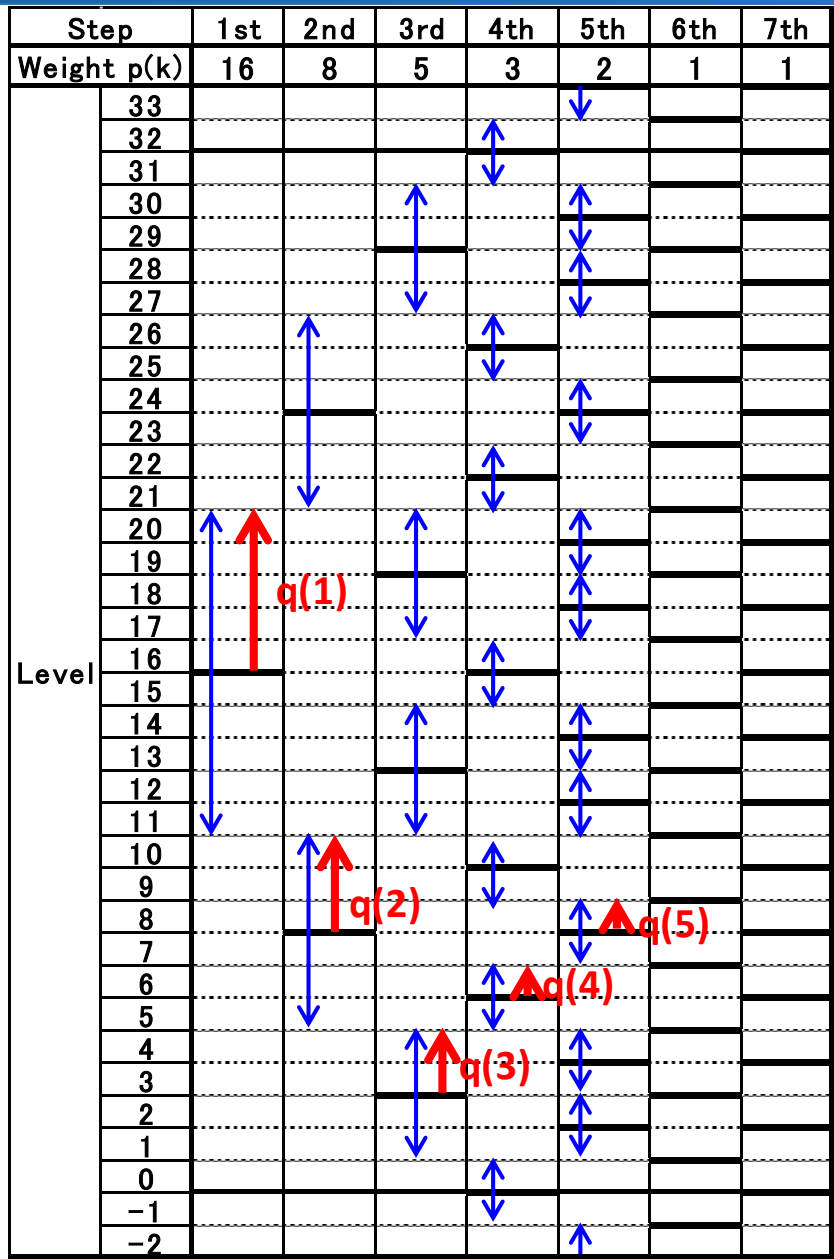
Two properties are discovered !!

1. *Correctable difference q(k) is always Fibonacci number* F_{M-k-1}.
2. ***q(k) is exactly in contact q(k+1) without overlap.***

If radix is **bigger** than 1.62
→ **separated** q(k)

If radix is **smaller** than 1.62
→ **overlapped** q(k)

Golden ratio can establish a standard !



Significance of Property 2

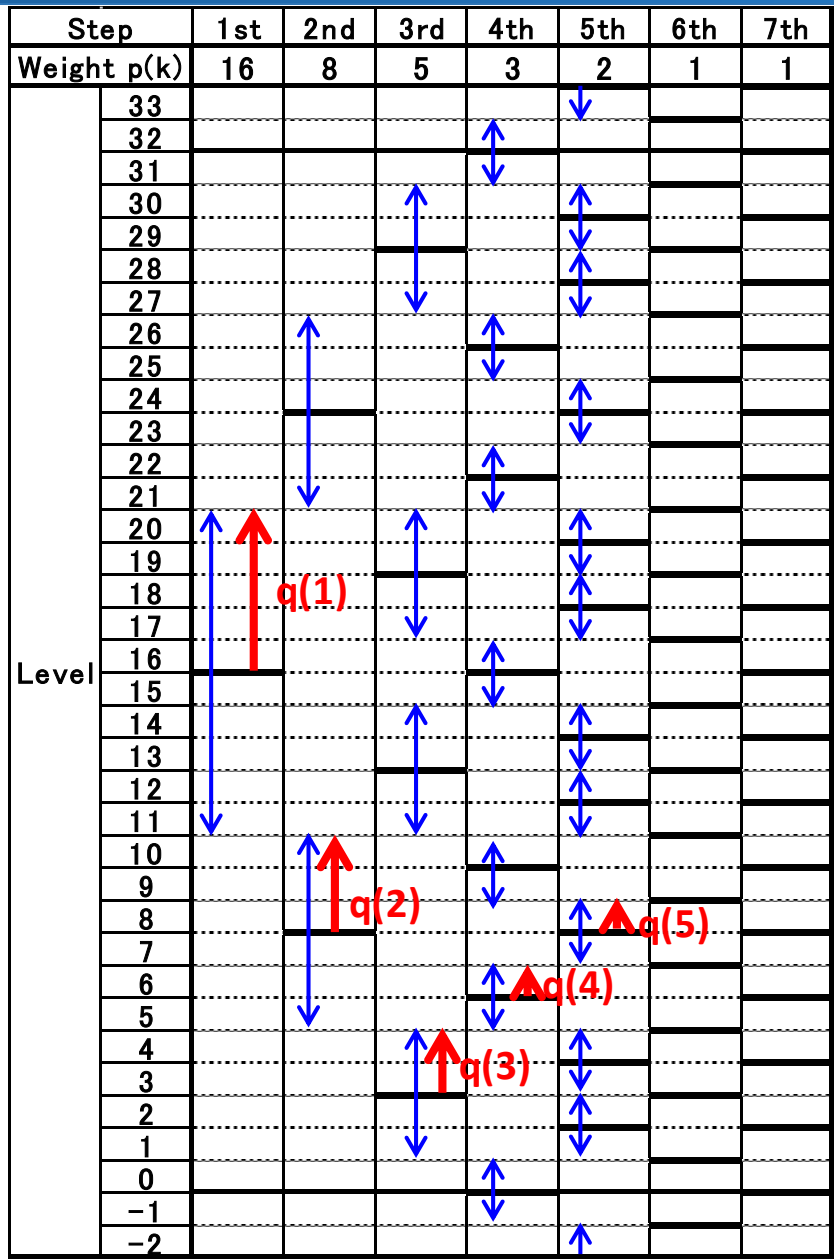
Fibonacci sequence SAR ADC

Two properties are discovered !!

1. *Correctable difference q(k) is always Fibonacci number F_{M-k-1}.*
2. ***q(k) is exactly in contact q(k+1) without overlap.***

Golden ratio covers wide input range by minimum extra comparison steps.

The most efficient design !

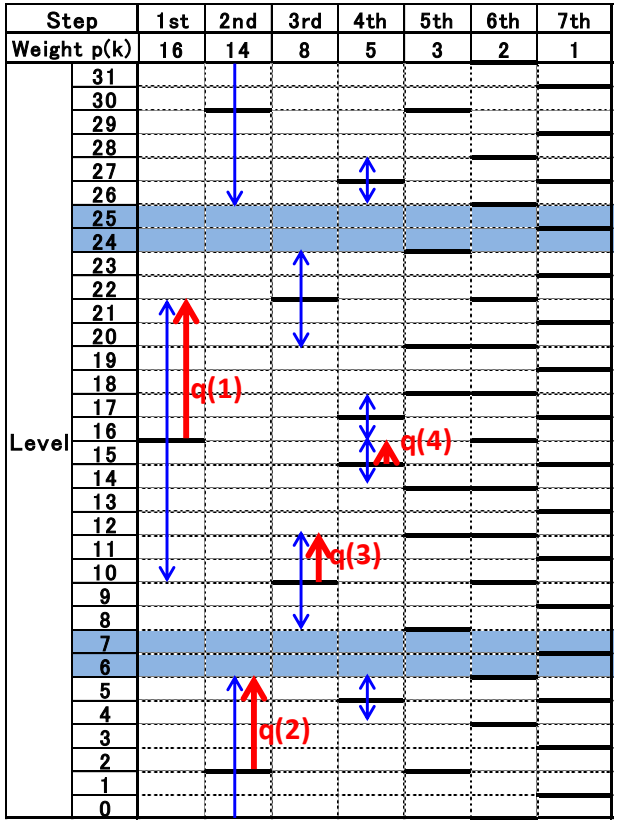


Comparison with Conventional Method

5bit SAR ADC

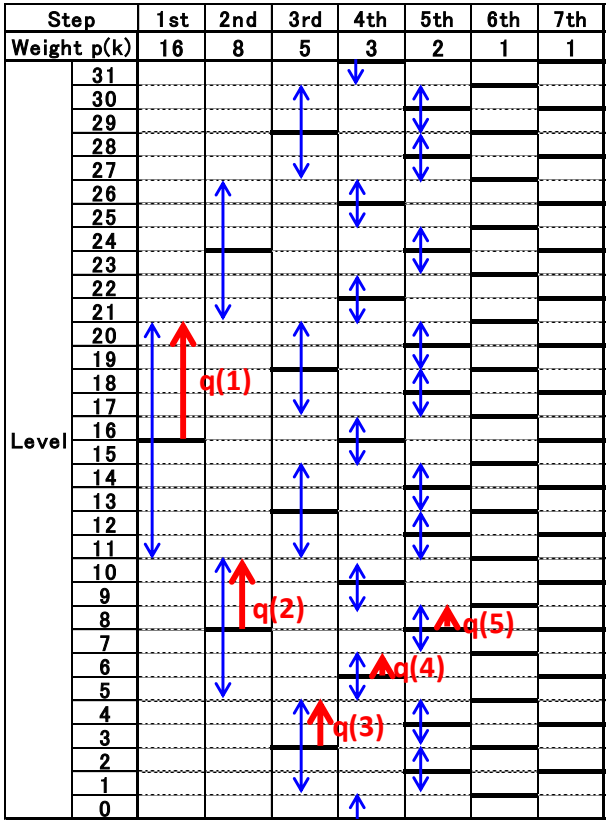
Conventional method

Radix=1.7



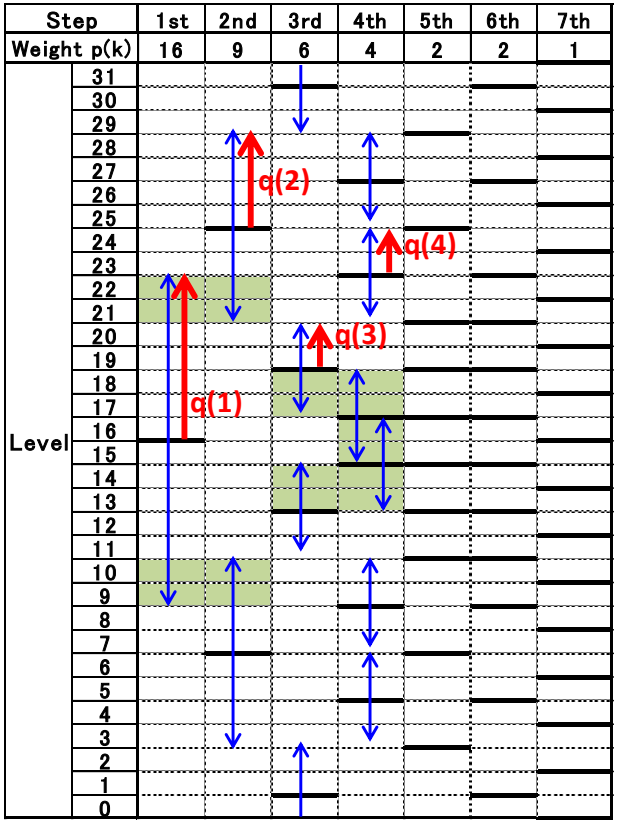
Proposed method

Radix=1.62



Conventional method

Radix=1.55



Redundant design using Fibonacci sequence

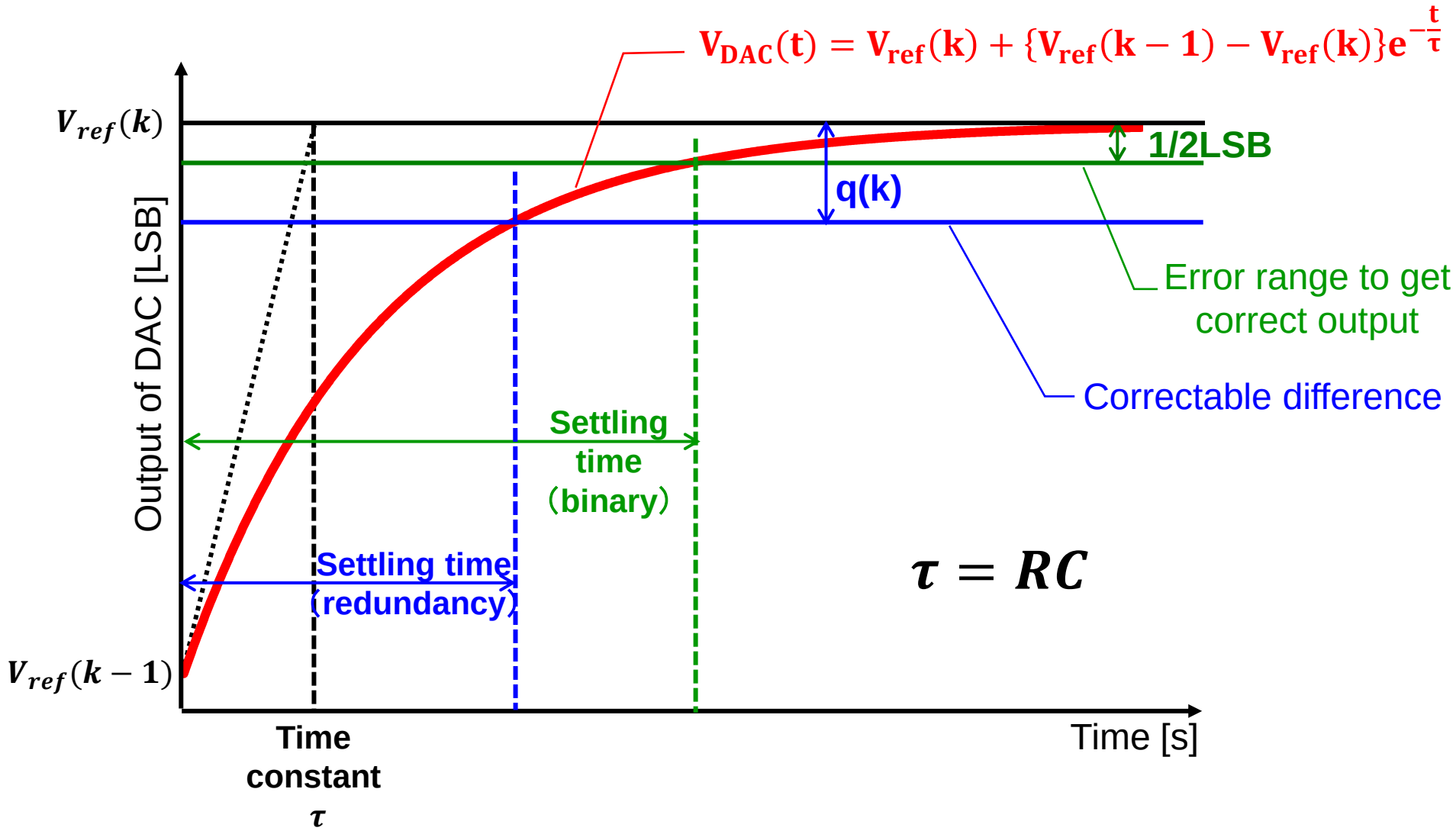
{ Radix standard
Efficient design

Outline

- Research Background
- SAR ADC Redundancy Design
- Proposed SAR Algorithm Using Fibonacci Sequence
- **Advantages of Proposed SAR Algorithm**
- Realization of Fibonacci DAC
- Conclusions

DAC Settling Time

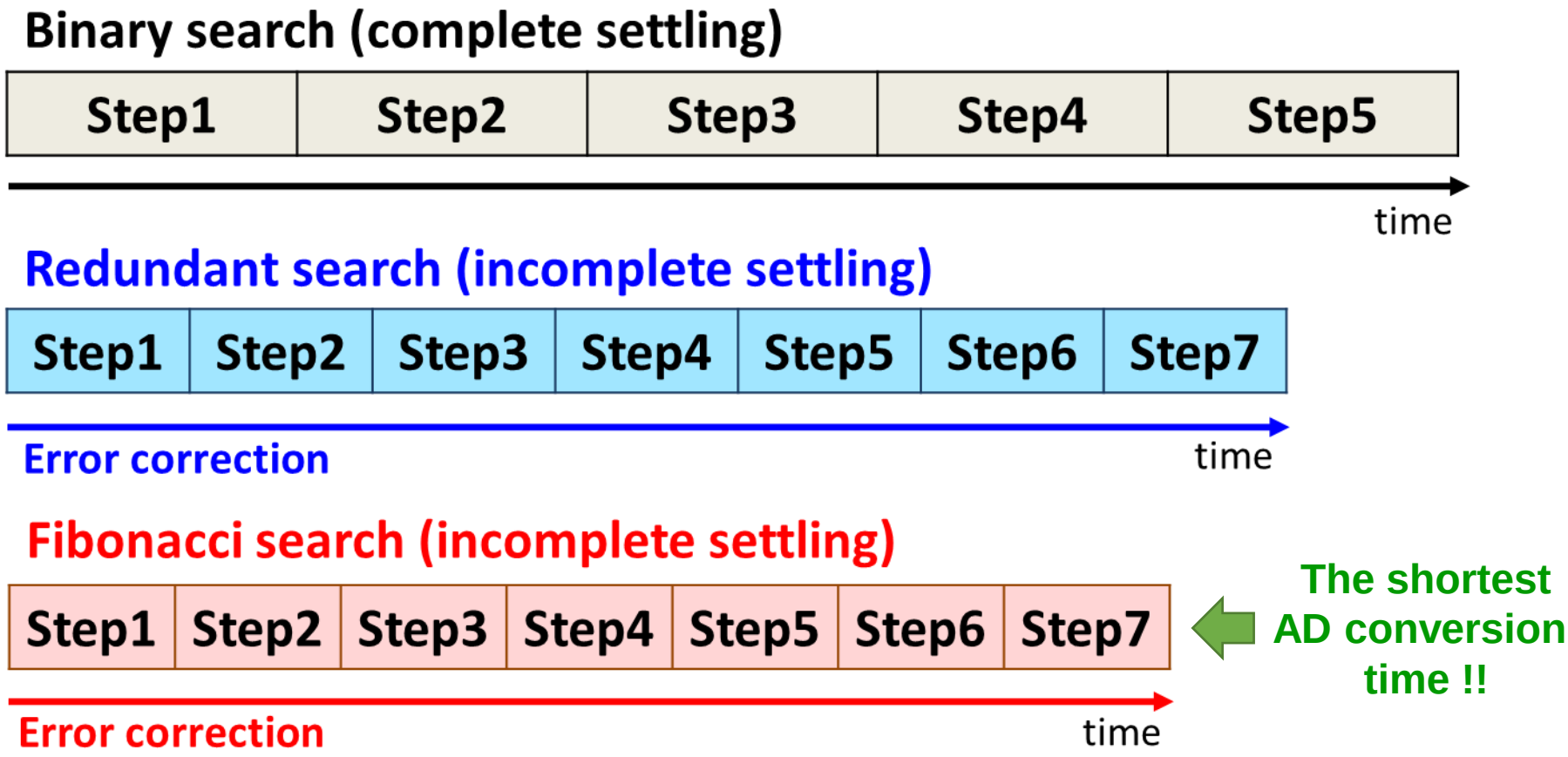
DAC Settling model by a simple first-order RC circuit



Reduction of Settling Time

Digital error correction ⇒ **Incomplete settling**

5bit SAR ADC

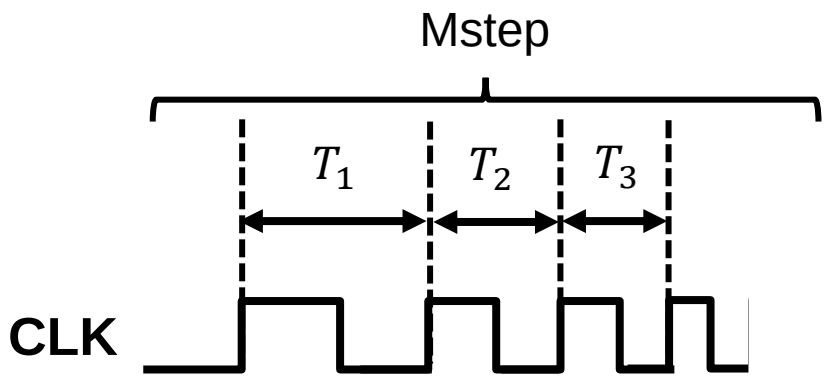


Generalization of DAC Incomplete Settling

Settling time

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

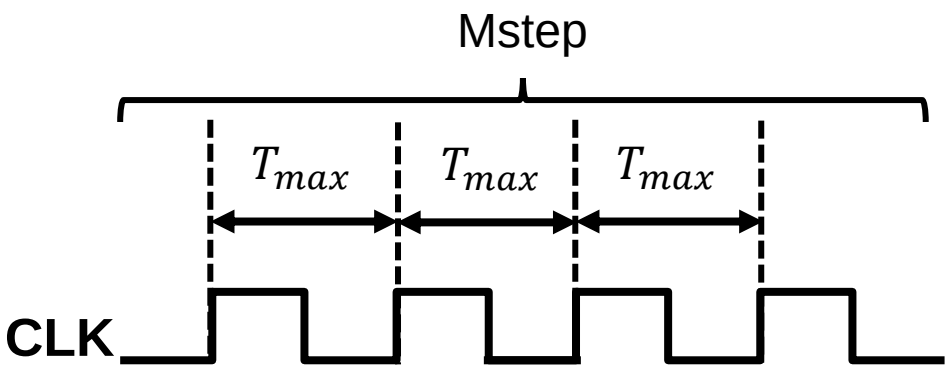
Variable clock SAR ADC



Total settling time

$$T_{total\_vari} = \sum_{i=1}^{M} T_i$$

Fixed clock SAR ADC

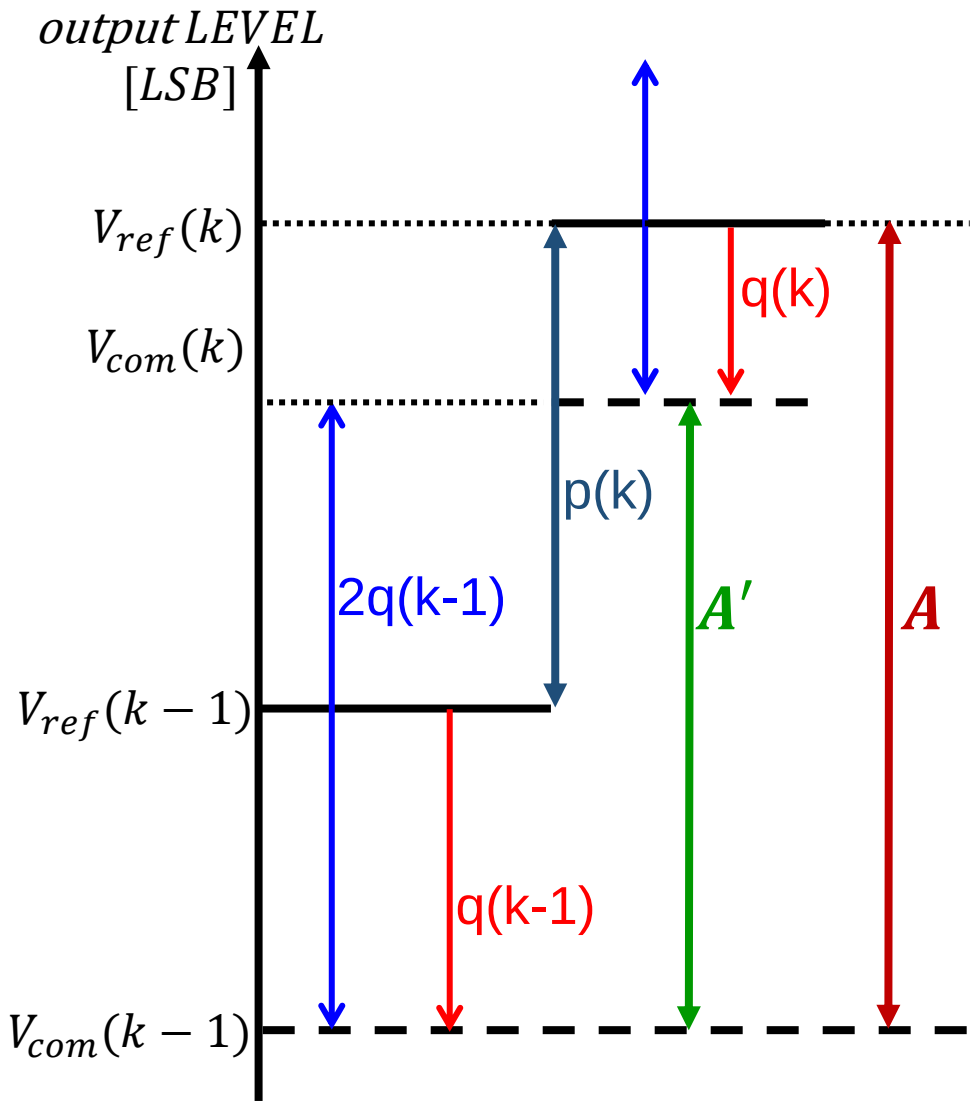


Total settling time

$$T_{total\_fixed} = T_{max} \times M$$

Settling Time in Fibonacci Case

Fibonacci sequence settling time



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

using Fibonacci sequence

$$p(k) = F_{M-k+1}$$
$$q(k) = F_{M-k-1}$$

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

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

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

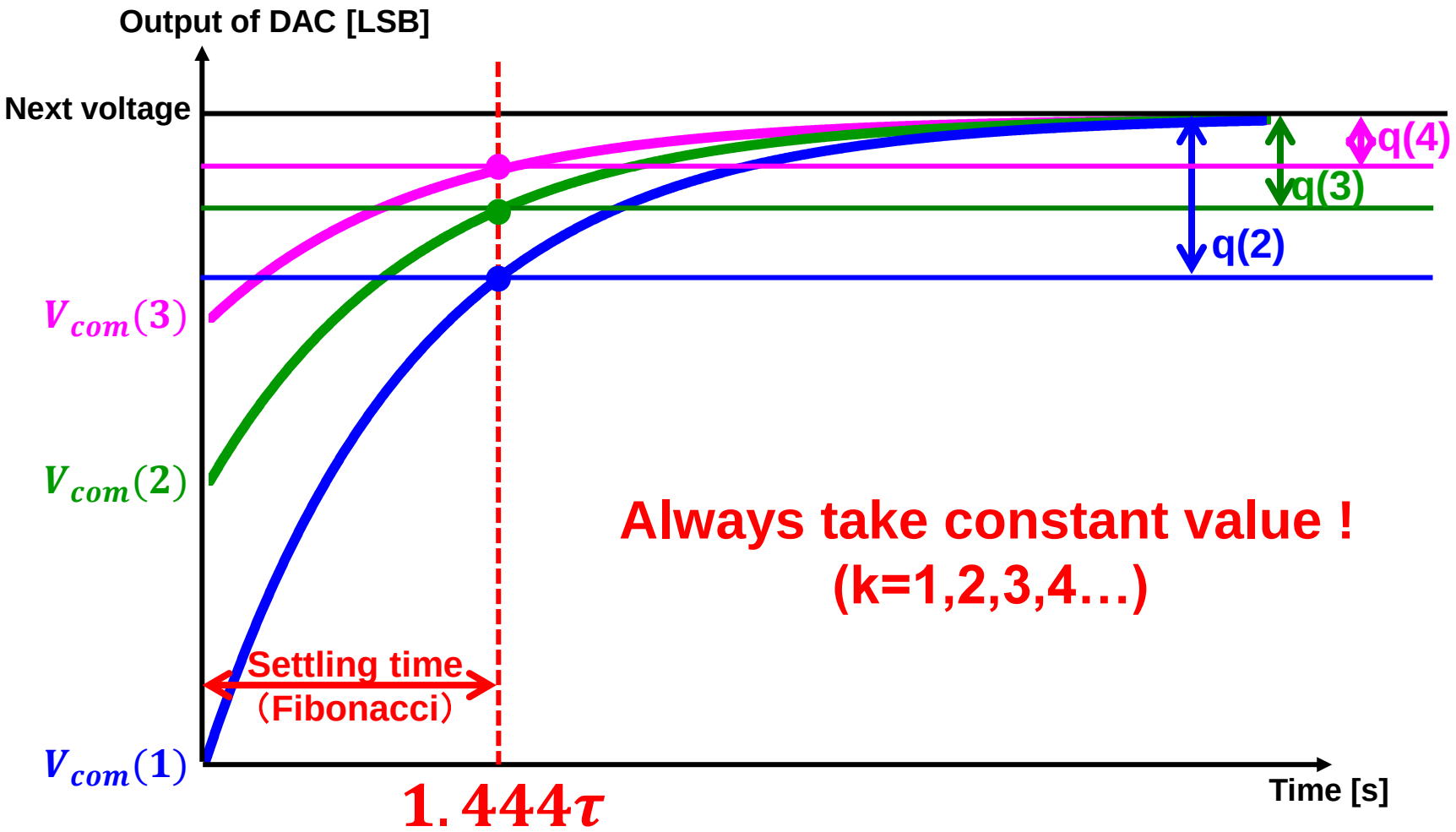
$$\lim_{n\to\infty}\frac{F_n}{F_{n-1}} = 1.618 = \varphi$$

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

Settling Time in Fibonacci Case

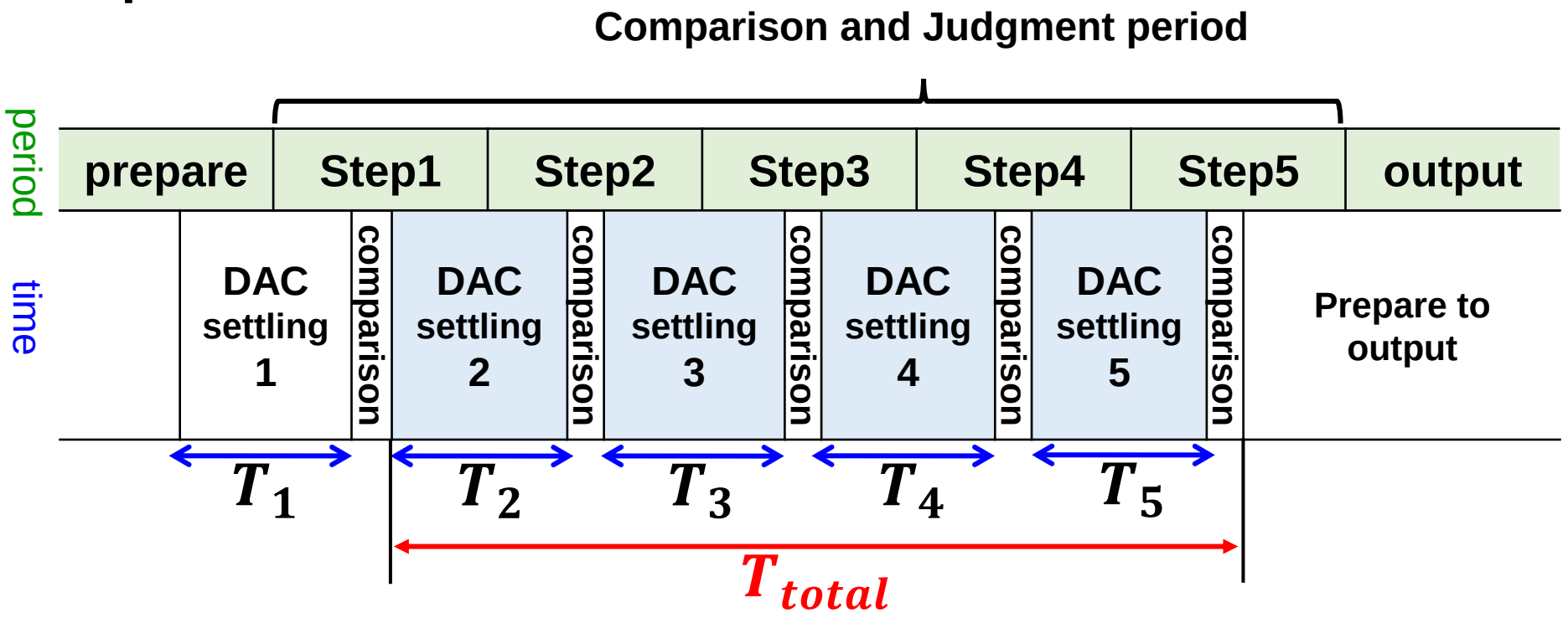
New property is discovered !

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



Simulation of Total Settling Time

5step SAR ADC

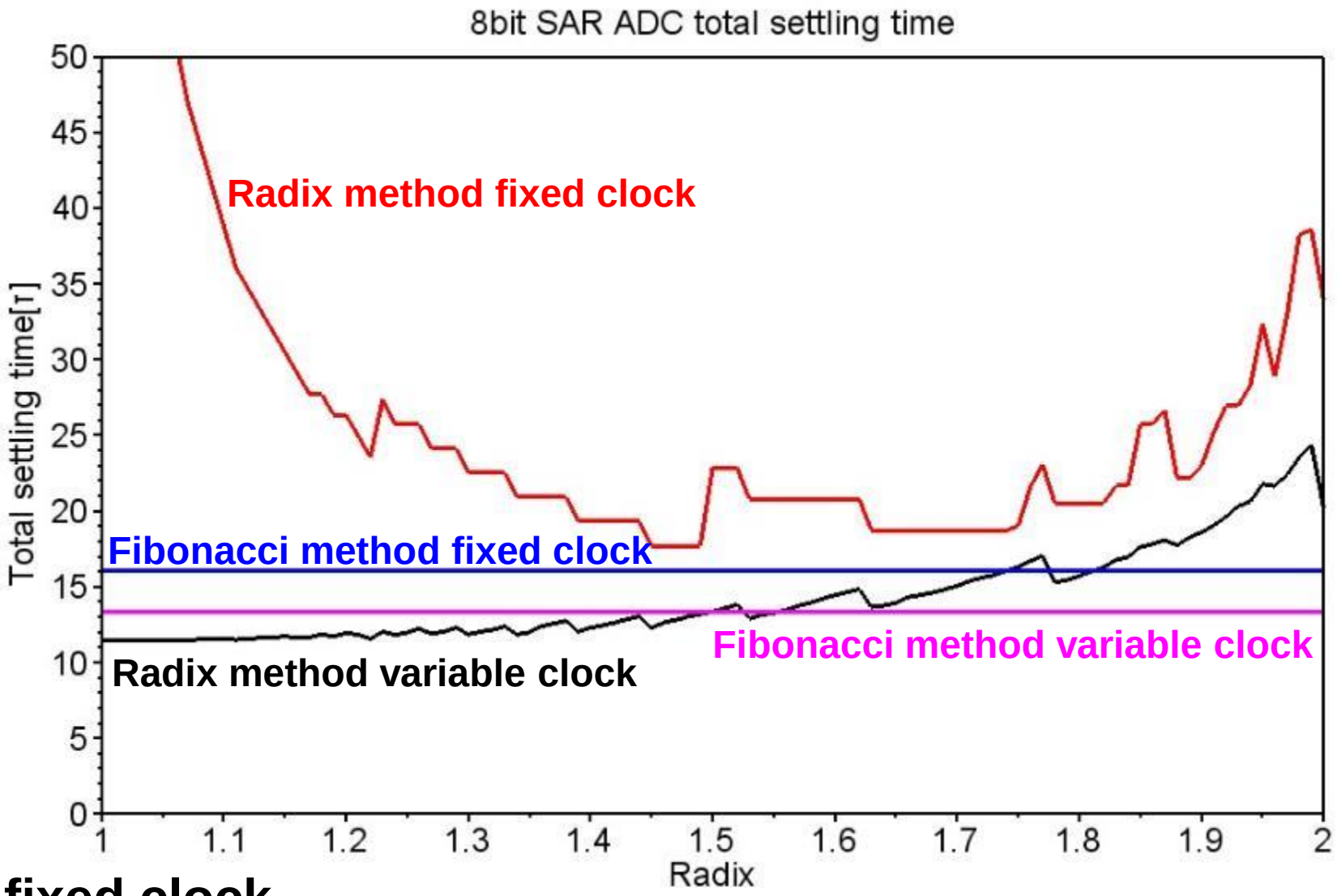


Conditions of simulation

- Investigate total settling time T_{total} of Variable clock and Fixed clock

$$T_{settle}(k) = \tau \ln\left(\frac{p(k)+q(k-1)}{q(k)}\right), \qquad T_{settle}(2) = \tau \ln\left(\frac{p(k)}{q(k)}\right)$$

Result of Fibonacci SAR ADC Settling

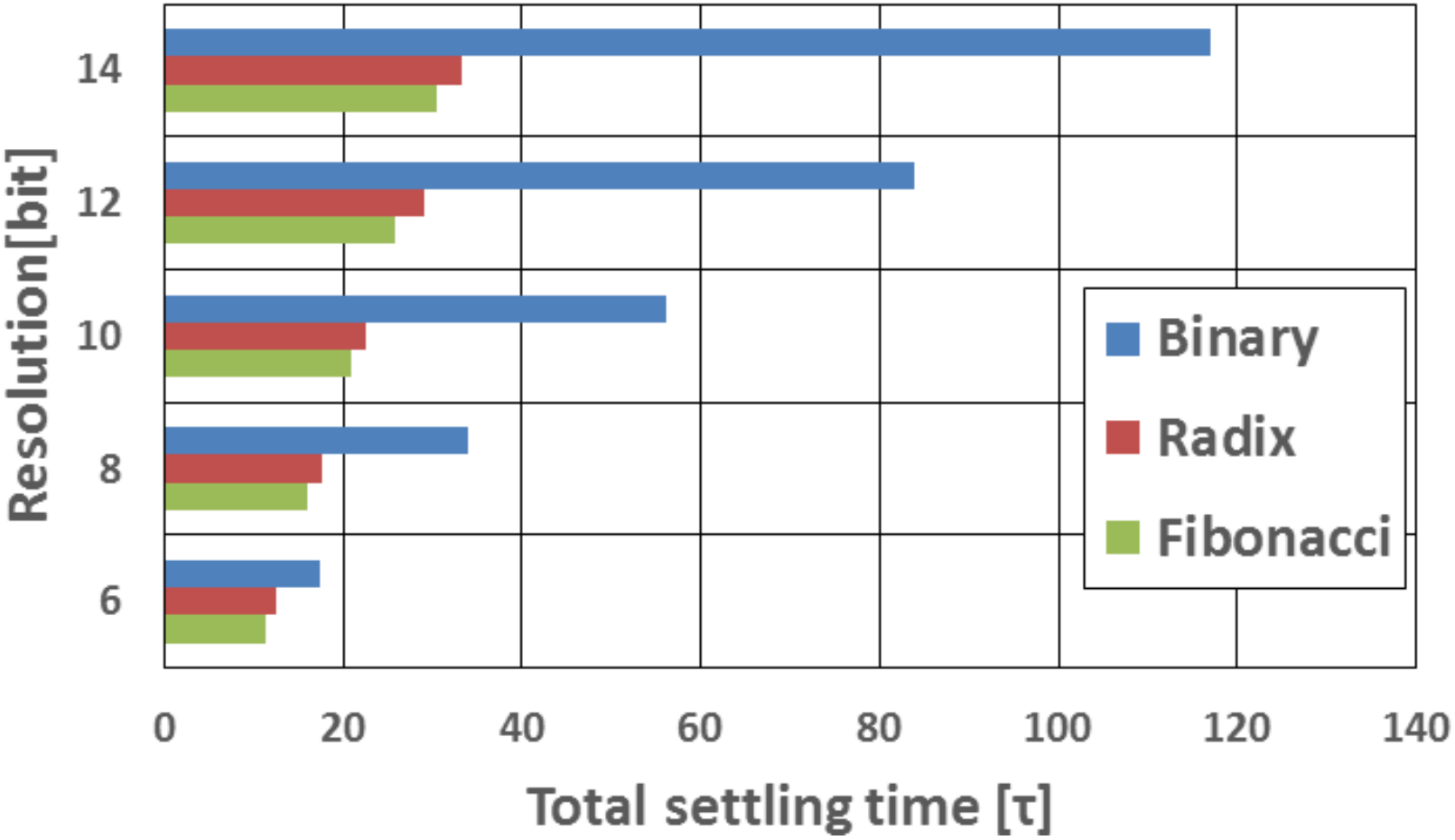


At fixed clock,

Fibonacci SAR ADC realizes the shortest settling time !!

Summary of Result

Result of each resolution at fixed clock



At fixed clock,

Fibonacci SAR ADC is faster than Radix SAR ADC !

Outline

- Research Background
- SAR ADC Redundancy Design
- Proposed SAR Algorithm Using Fibonacci Sequence
- Advantages of Proposed SAR Algorithm
- **Realization of Fibonacci DAC**
- Conclusions

Simplification of Redundant SAR ADC Circuit

Fibonacci redundancy design

⇒ High reliability and high speed conversion

Store reference voltage weight p(k)

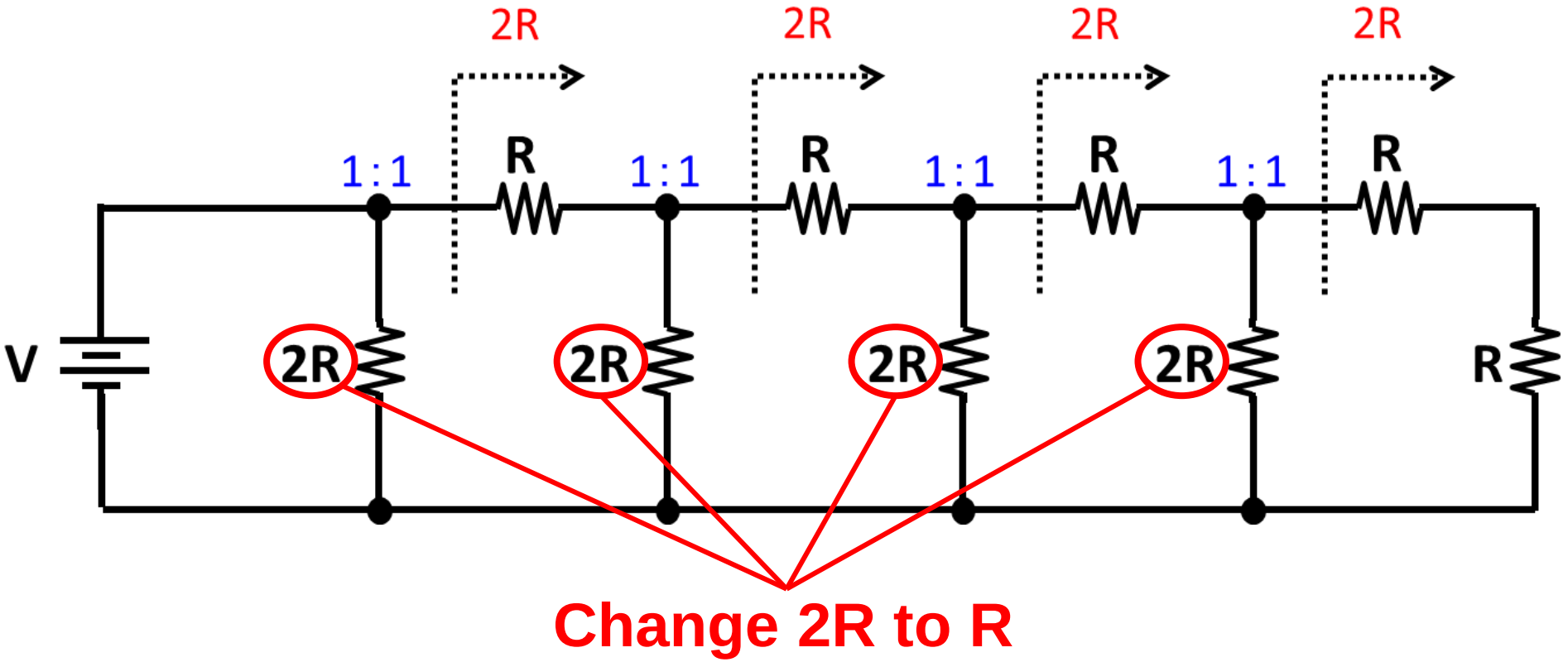
Conventional redundant SAR ADC

Try to change this DAC to **Fibonacci DAC** for easy design

R-2R Resistor Ladder

R-2R Resistor ladder network

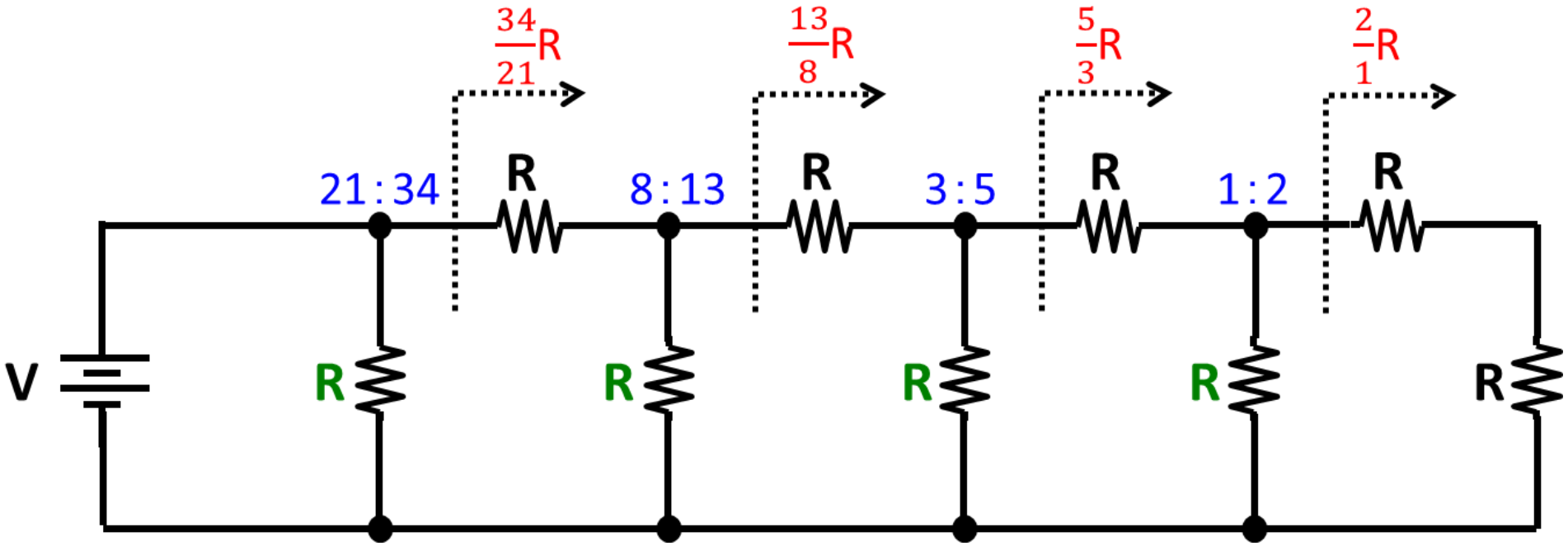
- Divides current into halves in each node
- Used for **binary** DAC



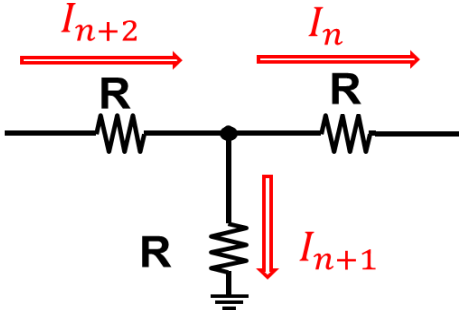
R-R Resistor Ladder

R-R Resistor ladder network

- Divides current into **Fibonacci ratio** in each node



Principle



$$I_{n+2} = I_{n+1} + I_n$$

$$\Updownarrow$$

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

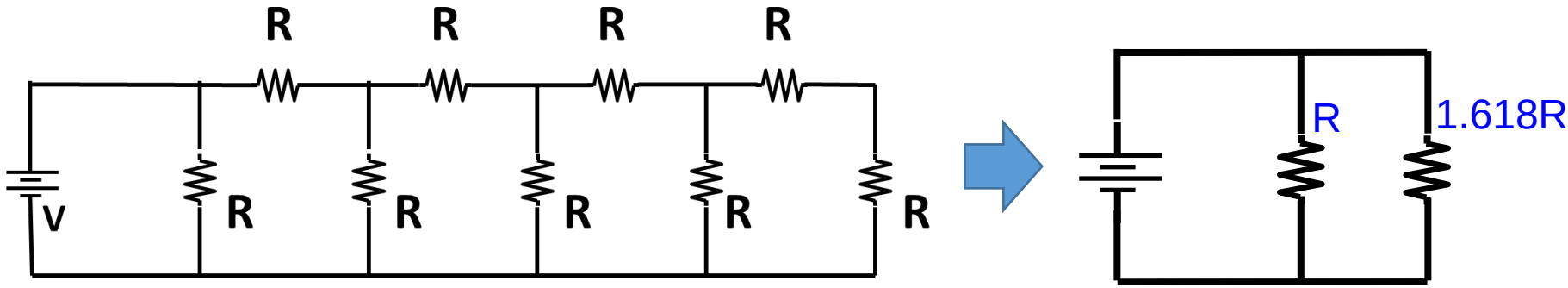
Golden Divide Resistor Ladder Network

R-R Resistor ladder network

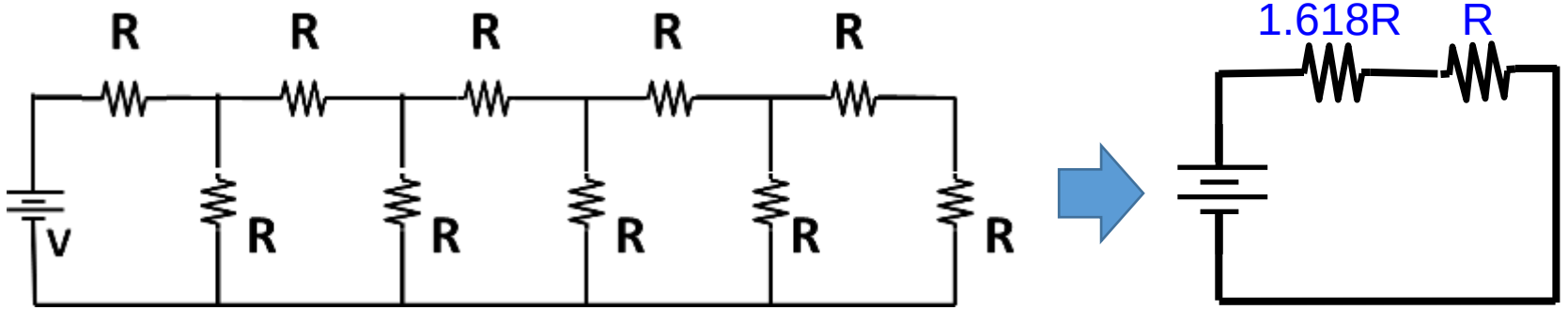
- Realize Golden divide of voltage or current
- High precision Golden divide

⇒Use for Fibonacci redundant SAR ADC

◆Current-dividing circuit



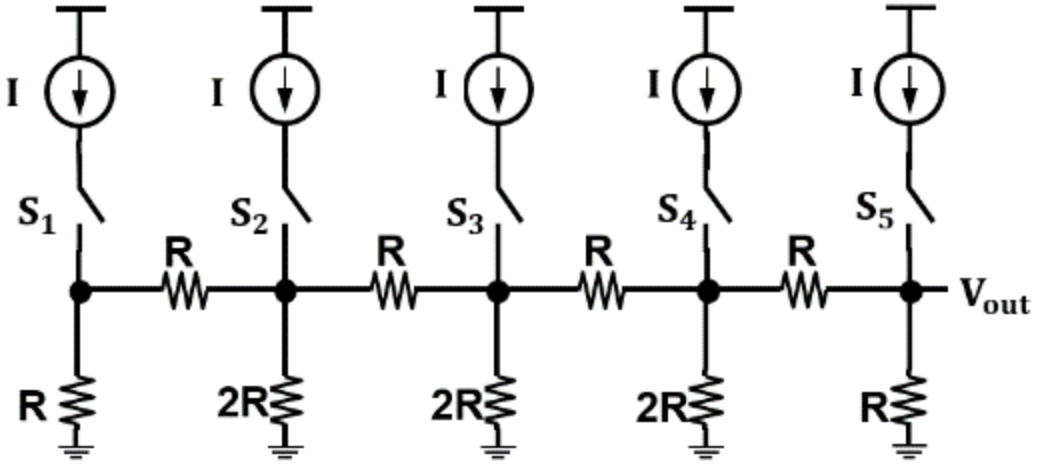
◆Voltage-dividing circuit



Proposal of Fibonacci DAC

R-2R resistor ladder

generates **binary** voltage



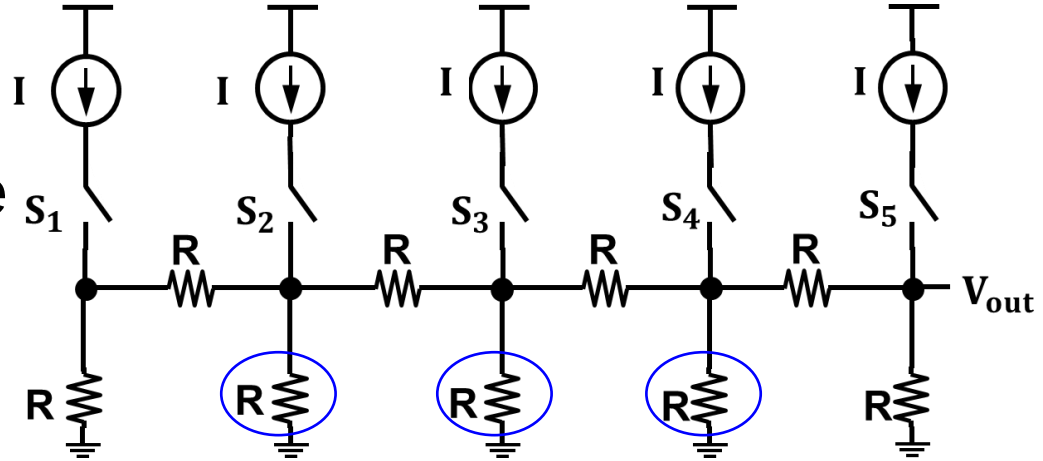
Only use R

Proposal

R-R resistor ladder

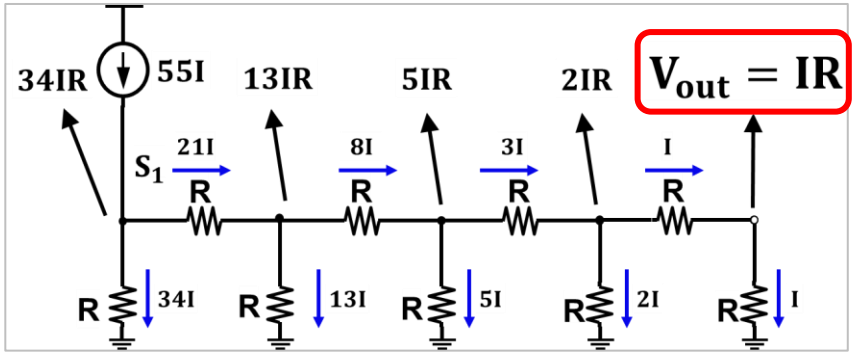
generates **Fibonacci** voltage

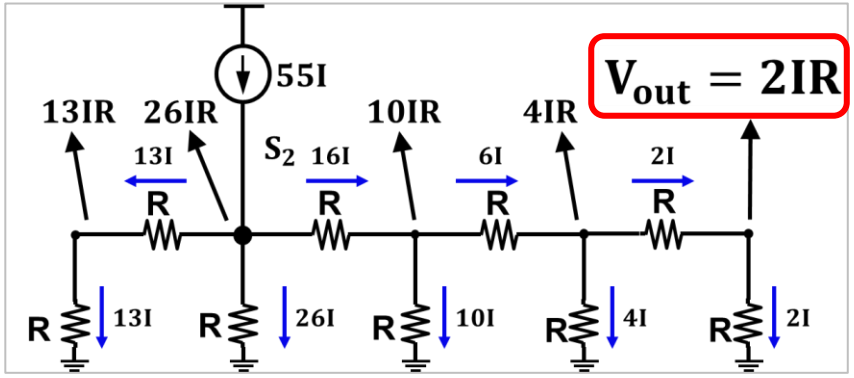
Realize Fibonacci DAC
by using simple circuit

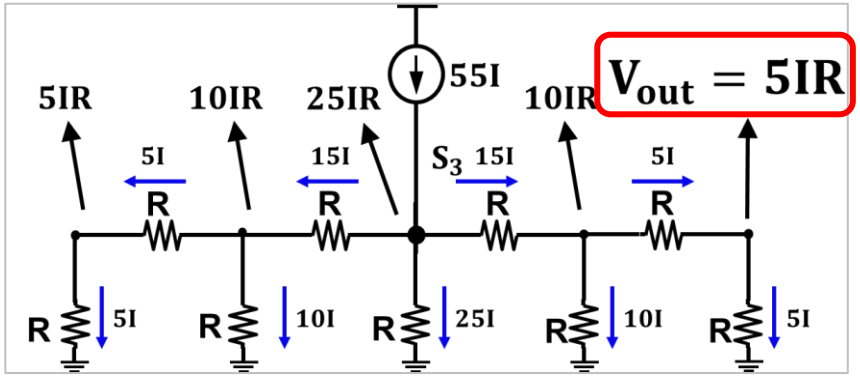


R-R resistor ladder network

Analysis of Fibonacci DAC







Output voltage

$$V_{out}(m) = \left(\frac{F_{2(N-m)+1}}{F_{2N}}\right) IR$$

N: the number of nodes

m: a connected node number from the left

Odd term of Fibonacci sequence

1, 1, **2**, 3, **5**, 8,**...**

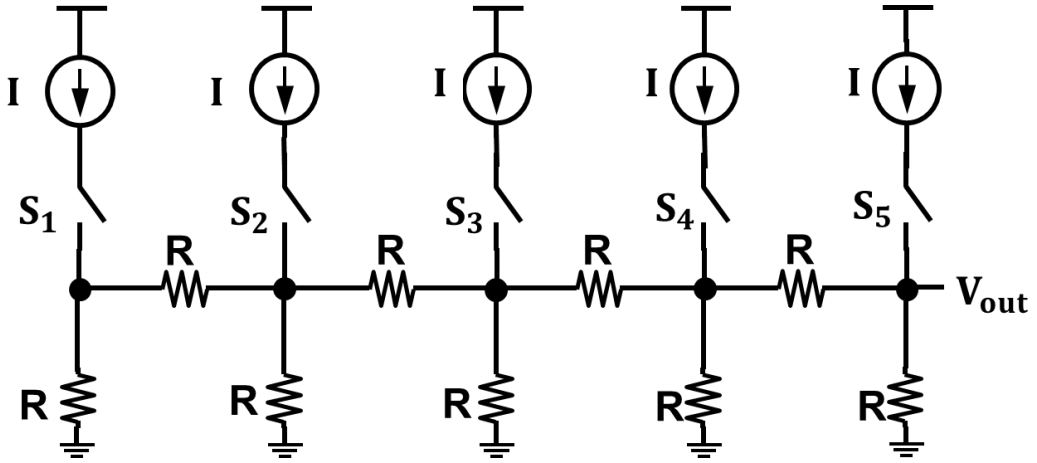
We also need even term

Proposal of R//R Fibonacci DAC

R-R resistor ladder

Generate

Fibonacci voltage of **odd** term



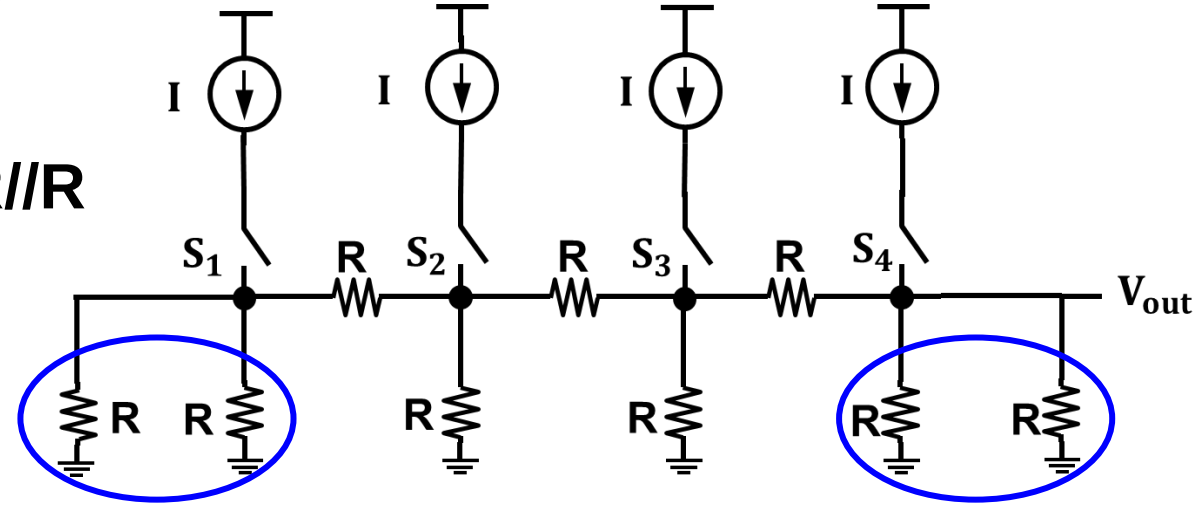
Change terminal resistor to parallel resistors

Proposal

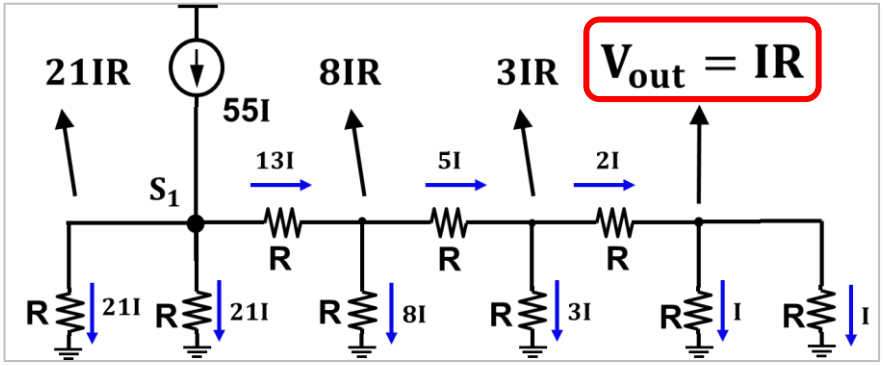
R-R resistor ladder with terminations of R//R

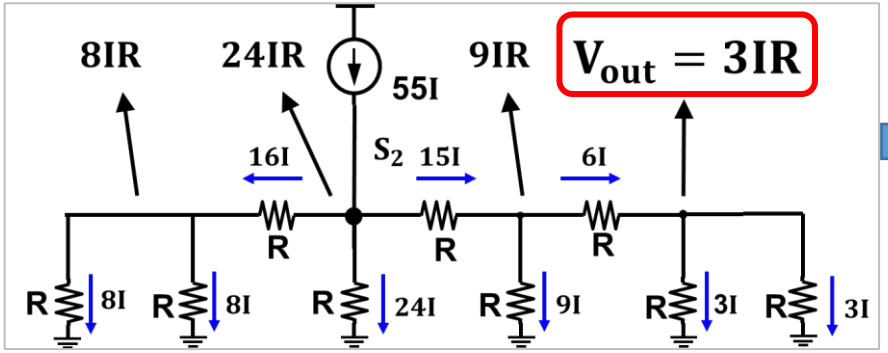
Generate

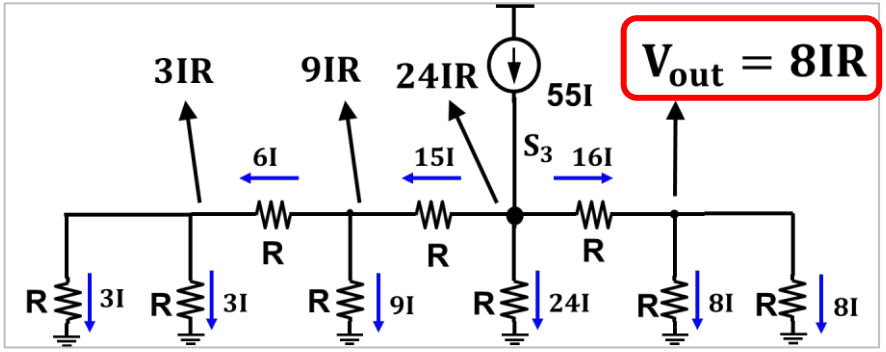
Fibonacci voltage of **even** term



Analysis of Fibonacci DAC







Output voltage

$$V_{out}(m) = \left(\frac{F_{2(N-m+1)}}{F_{2(N+1)}}\right) IR$$

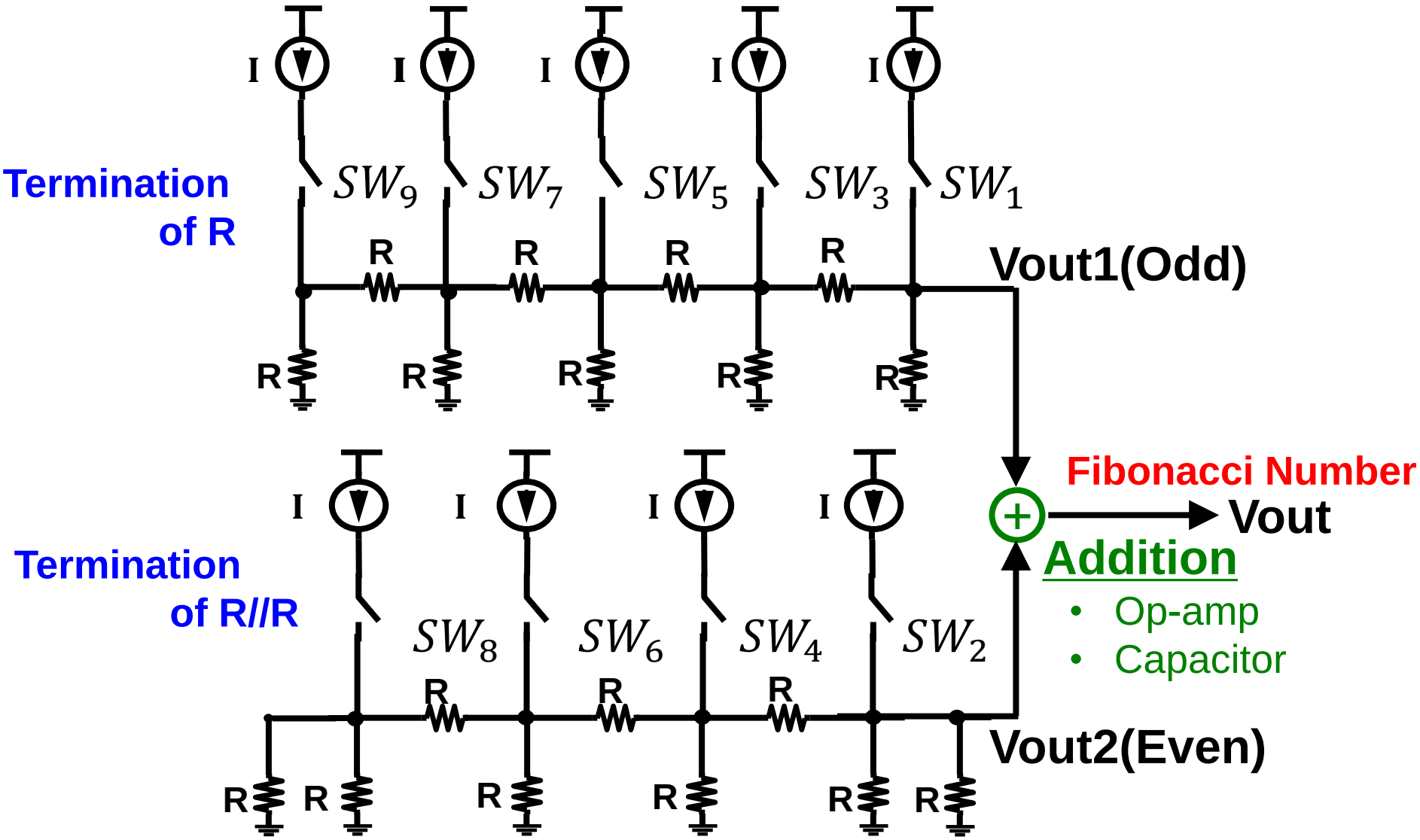
N: the number of nodes

m: a connected node number from the left

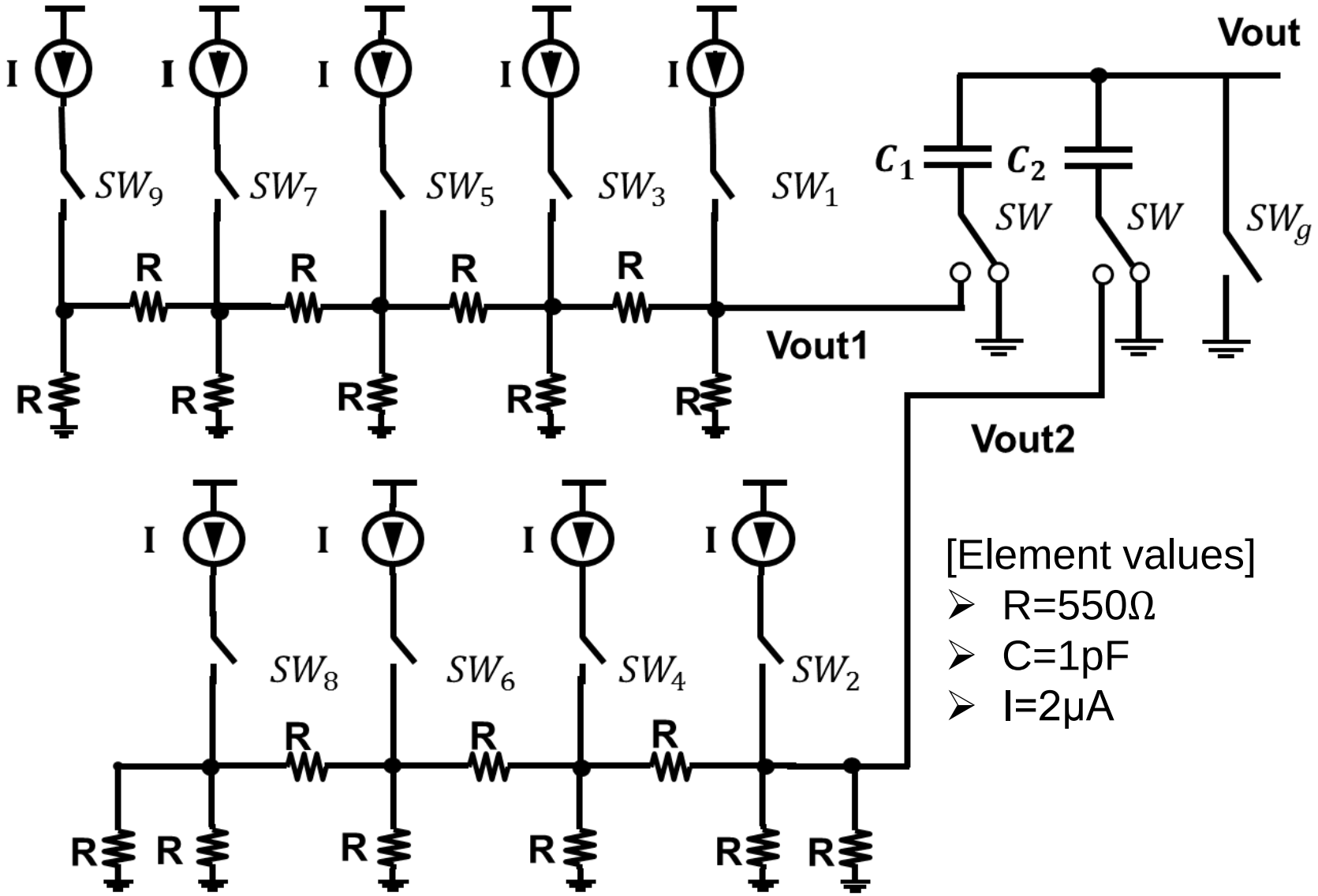
Even term of Fibonacci sequence

1, **1**, 2, **3**, 5, **8**,...

Fibonacci DAC



Fibonacci DAC Simulation



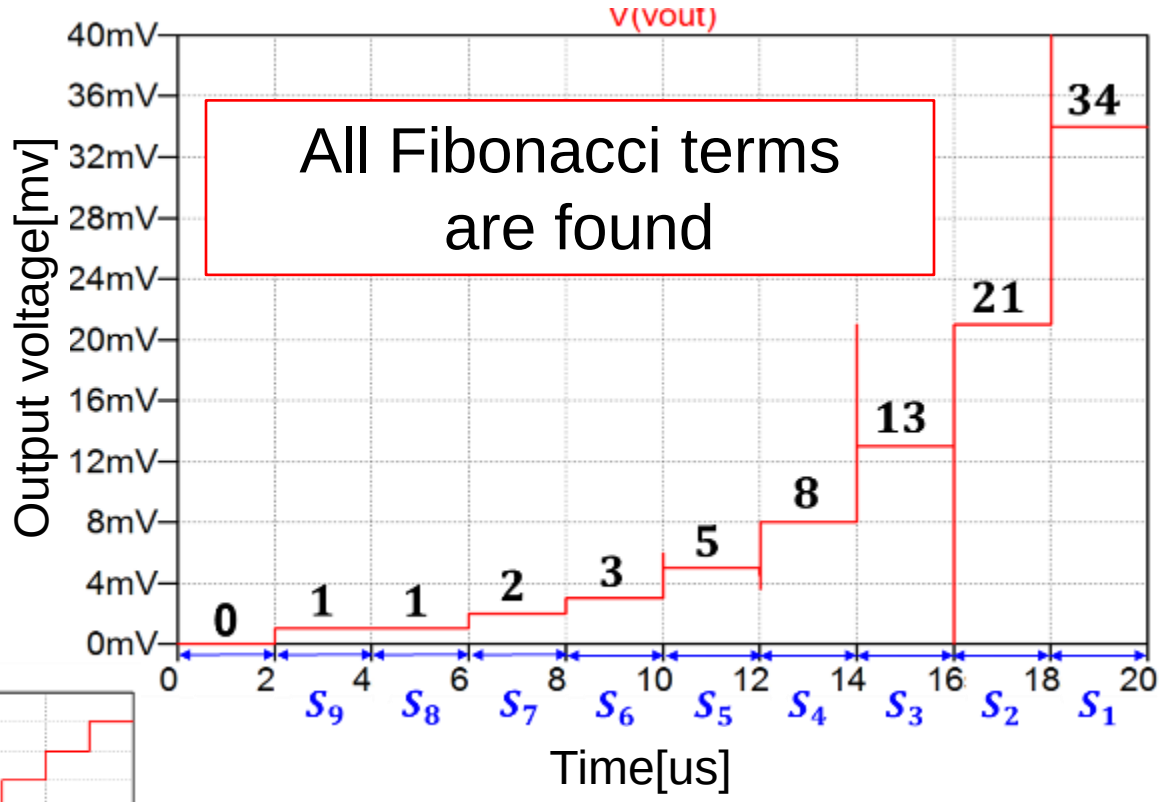
[Element values]

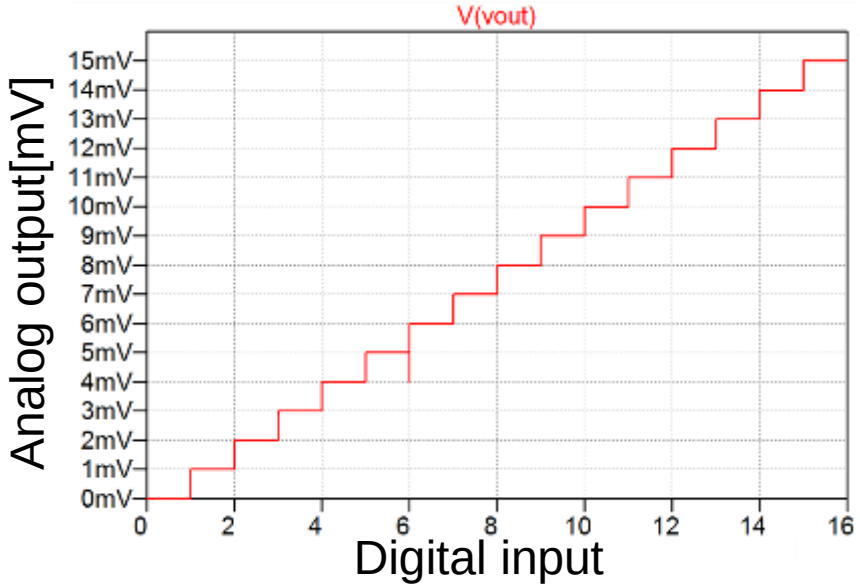
- R=550Ω
- C=1pF
- I=2μA

Simulation Result

Operation simulation

Each switch corresponds to a Fibonacci term





A-D conversion simulation

Combination of current sources realizes DAC function

Fibonacci DAC is realized

Conclusion

- Propose redundant SAR ADC design methods

- Get important properties by using Fibonacci sequence
  - **Reliable**
    Correctable difference covers wide input range
  - **Shortest-Conversion**
    Conversion time is shortest in a fixed clock
  - **Radix-Standard**
    Golden ratio φ establish radix standard
  - **Constant-Settling**
    DAC settling time is constant for each step

- Propose new DAC which can generate Fibonacci number voltage

Number Theory for ADC/DAC Design

Carolus Fridericus Gauss
(1777-1855)

"Number theory is the queen of mathematics"

Carolus Fridericus Gauss

Past number theory

Beautiful and mysterious
but NEVER practical

Present number theory

Used for information communication processing
➡ good match to digital technology

Our position

Number theory application for ADC/DAC is a frontier.
There are great chances for new discovering !

Kobayashi
Laboratory

Q & A

Q&A

◆フィボナッチDACのミスマッチはどうするのか？(抵抗や容量のばらつきはどうする？)

⇒内部DACの正確さはSAR ADCに重要である。まだミスマッチについては検討中であるので、今後慎重に検討していきたい。

◆実装するのか？(良い結果を期待しているよ)

⇒理論の正しさを実証するため、実装も考えていきたい。

◆他の整数論は試してみたのか？(何が一番良い？)

⇒他の整数論は「リュカ数列」「トリボナッチ数列」「Nボナッチ数列」を検討したが、フィボナッチ数列が最も良い結果となった。ただしリュカ数列はフィボナッチと同等の性質をいくつか見せることがある。

◆他の整数論を使うときはDACをどうすればよいのか？

⇒温度計コードを利用したDACを使う従来手法のメモリ(RAM)の内容を変更すればよい。しかし今回提案したフィボナッチDACが実現すれば、更なる小面積化を実現することができるかもしれない。