Optimization of Segmented DAC Linearity Improvement Algorithm Using Unit Cell Sorting with Digital Method

Yi Liu , Anna Kuwana ,
Shogo Katayama , Xiongyan Li ,
Atsushi Motozawa , Haruo Kobayashi

Division of Electronics and Informatics,
Gunma University
Renesas Electronics Corp

OUTLINE

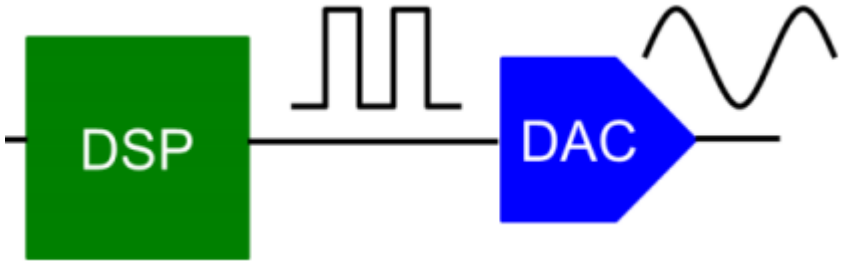
- Research Background and Objective
- Segmented Current-Steering DAC
- Problem Formulation
- Unit Current Cell Sorting Algorithm
- Simulation Results
- DAC Architecture with Sorting Algorithm
- Conclusion

OUTLINE

- Research Background and Objective
- Segmented Current-Steering DAC
- Problem Formulation
- Unit Current Cell Sorting Algorithm
- Simulation Results
- DAC Architecture with Sorting Algorithm
- Conclusion

Research Background

Digital-to-Analog Converter (DAC)



- Key component in modern transmitter circuits.
- High linearity is required.

Research Objective

Nano-CMOS implementation of DAC

- Device mismatch is large
- DAC linearity deteriorates
- Digital circuit can be implemented with small chip area

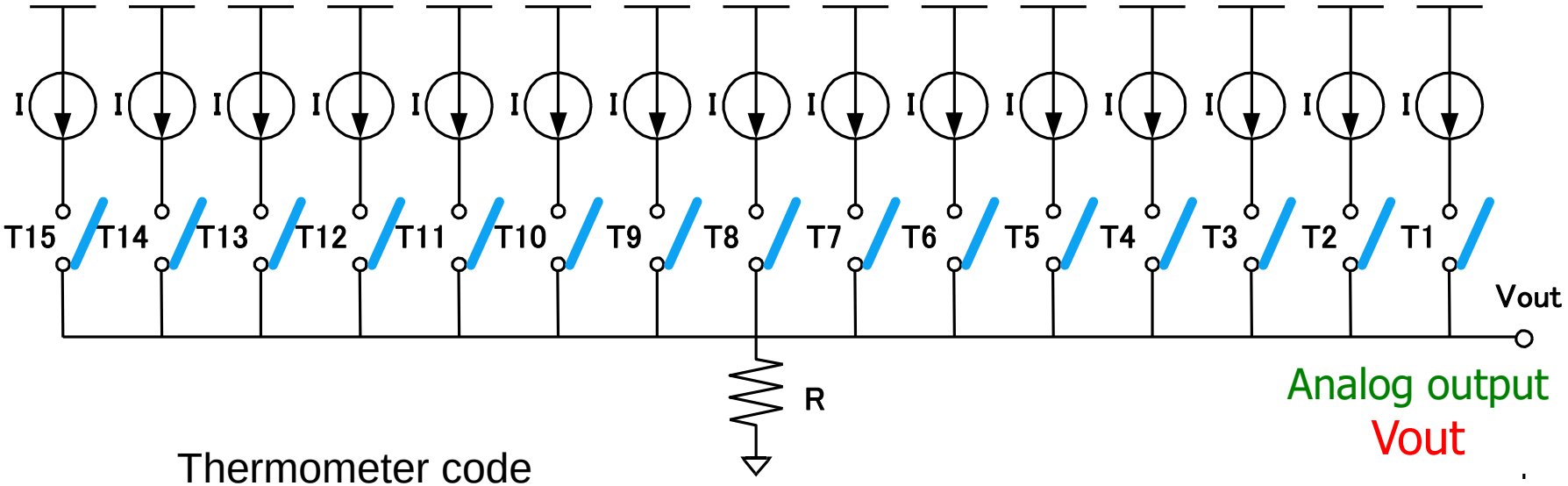
Development of digital calibration method for DAC non-linearity

Digitally-assisted analog technology

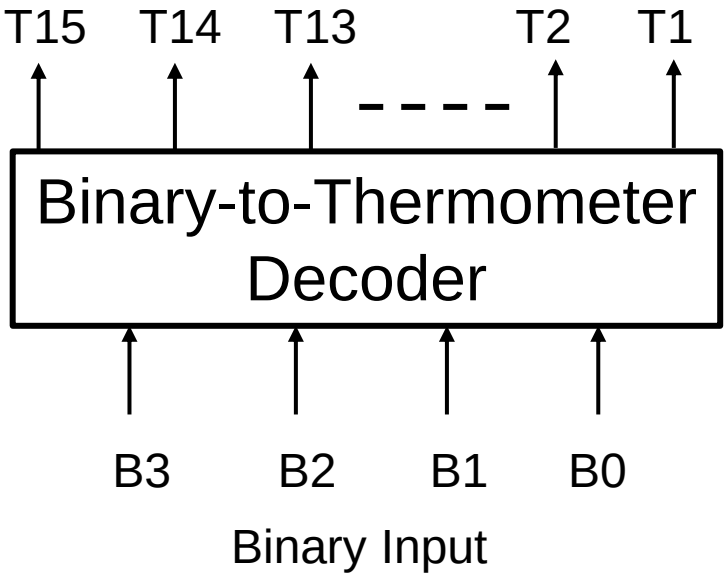
OUTLINE

- Research Background and Objective
- Segmented Current-Steering DAC
- Problem Formulation
- Unit Current Cell Sorting Algorithm
- Simulation Results
- DAC Architecture with Sorting Algorithm
- Conclusion

Segmented DAC Configuration: 4-bit case



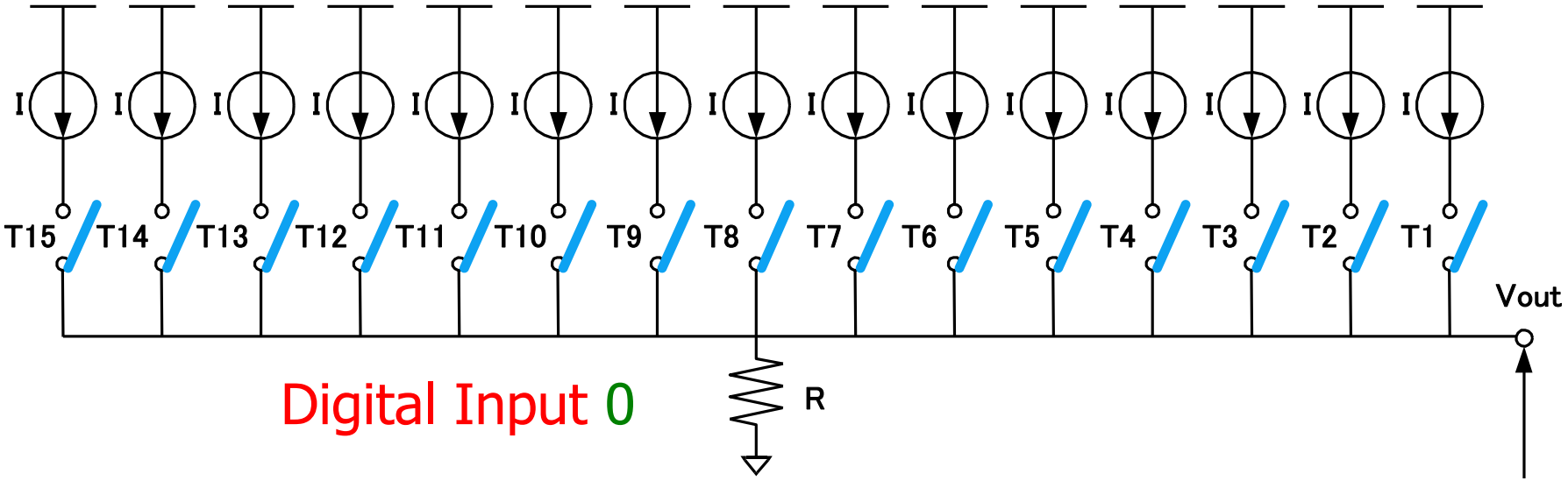
Thermometer code



Features

- Monotonicity guaranteed
- Small glitch
- Small DNL
- Decoder required

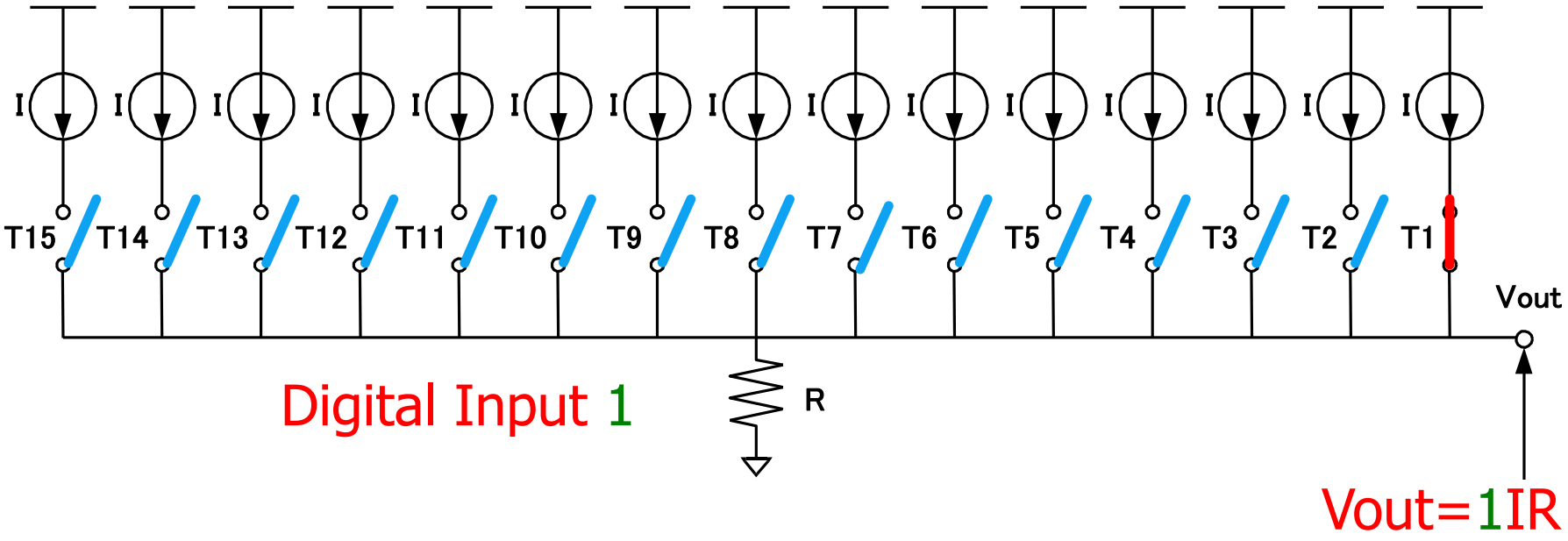
Segmented DAC Operation: Digital Input = 0



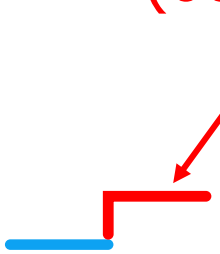
Vout=0

(0000000000000000)

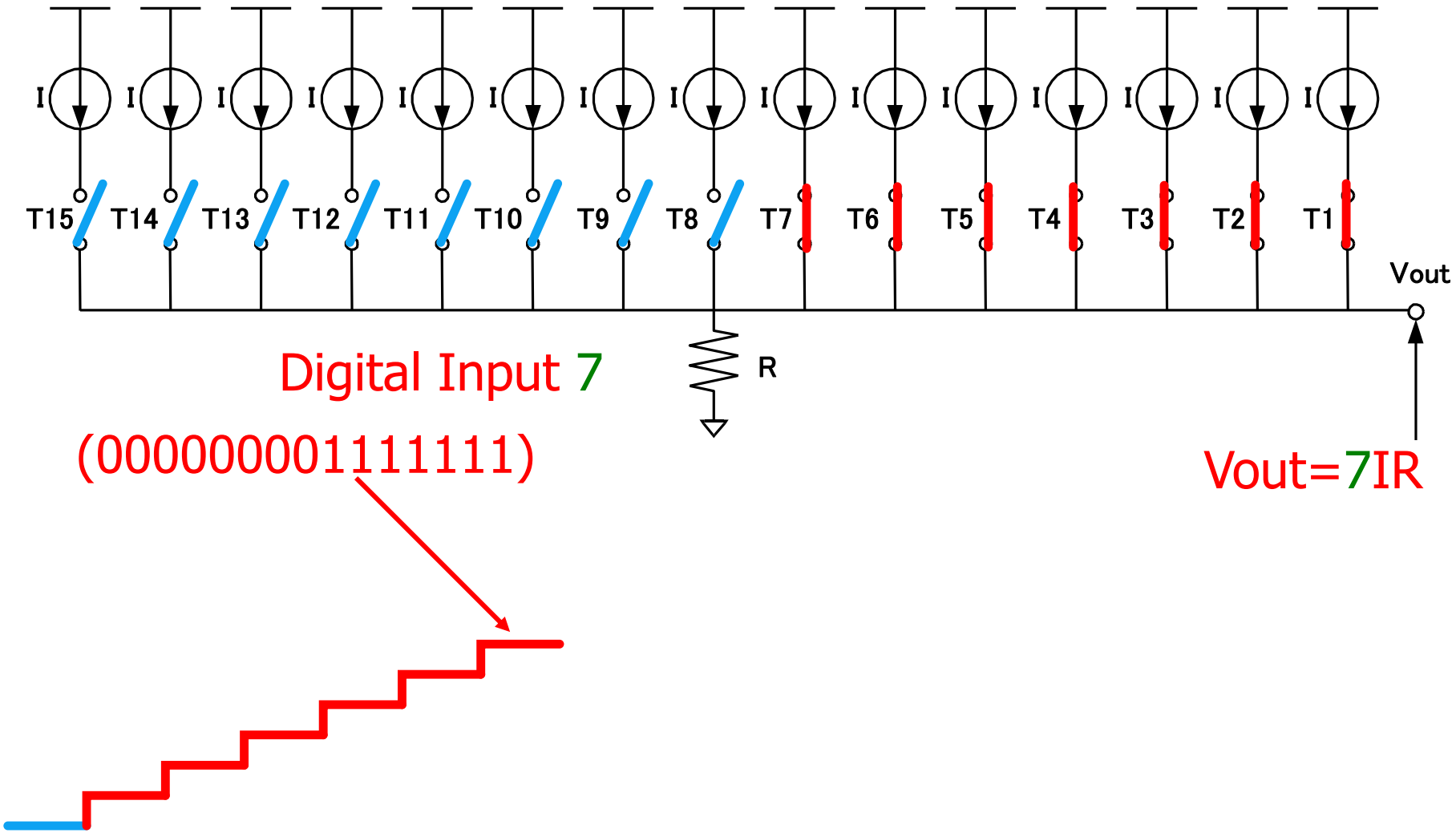
Segmented DAC Operation: Digital Input = 1



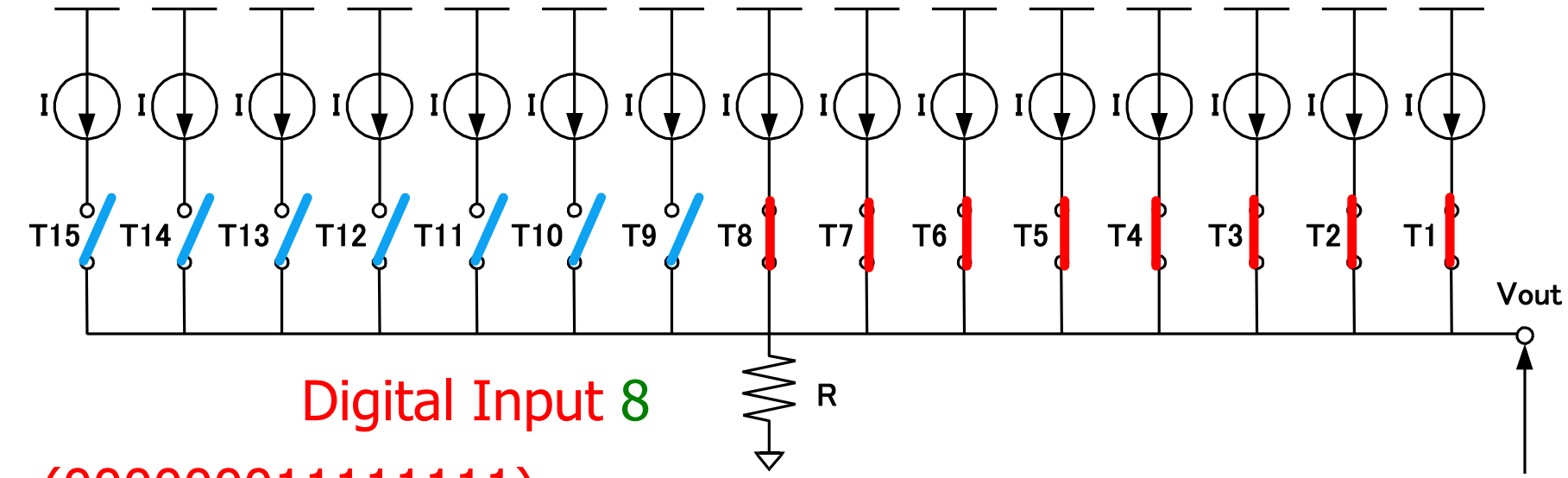
(0000000000000001)

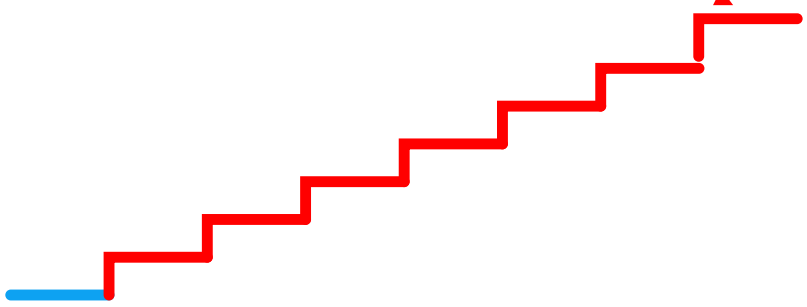


Segmented DAC Operation: Digital Input = 7

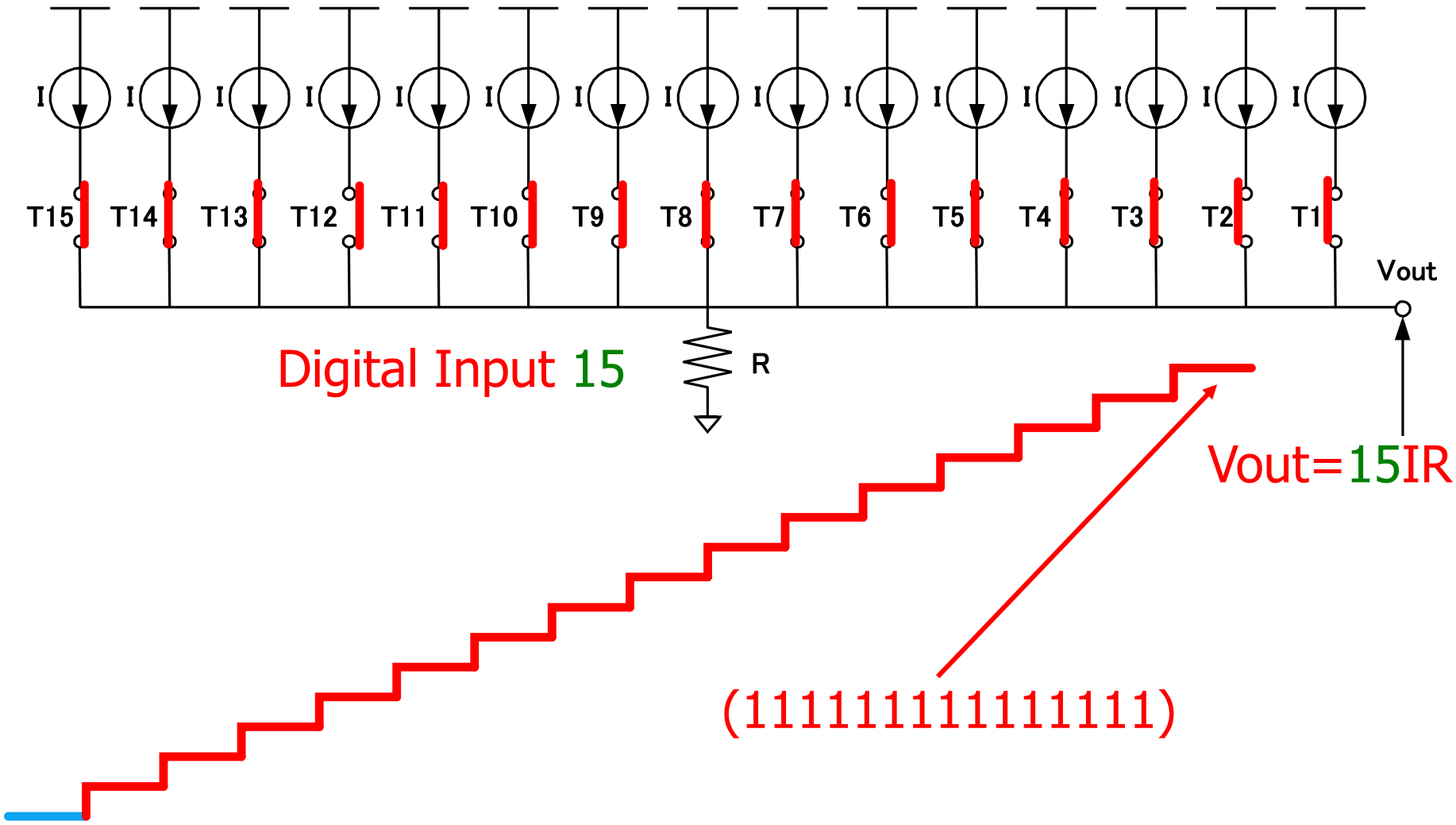


Segmented DAC Operation: Digital Input = 8





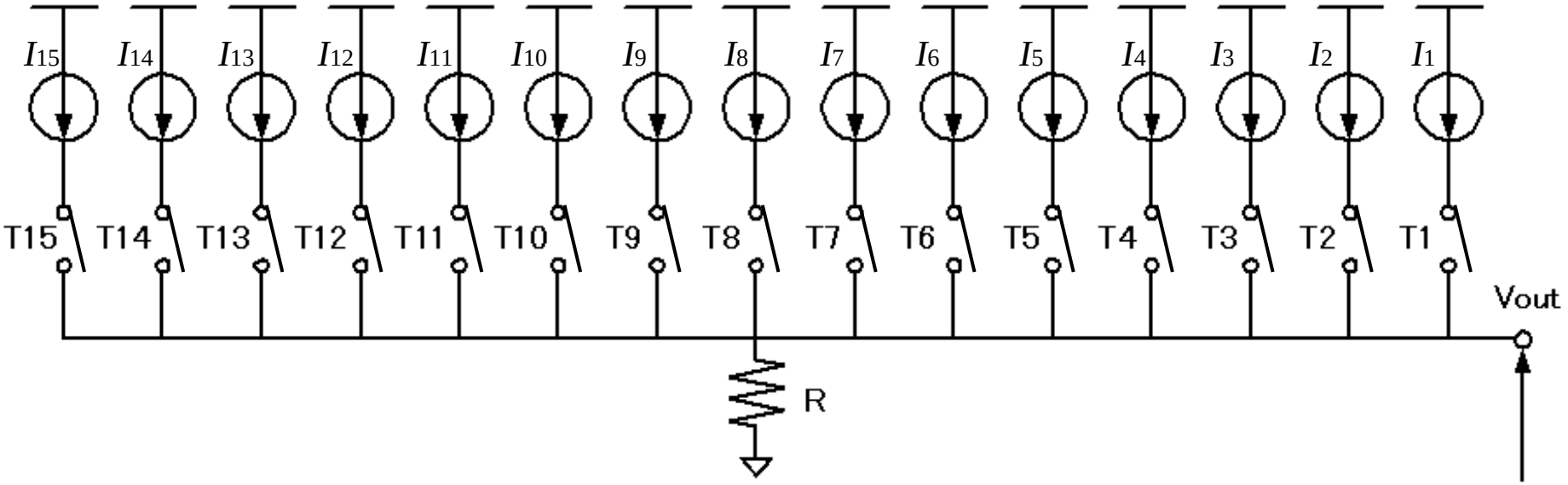
Segmented DAC Operation: Digital Input = 15



OUTLINE

- Research Background and Objective
- Segmented Current-Steering DAC
- Problem Formulation
- Unit Current Cell Sorting Algorithm
- Simulation Results
- DAC Architecture with Sorting Algorithm
- Conclusion

Problem of Actual Segmented DAC



Ideal DAC :

$$I_1 = I_2 = I_3 = \cdots = I_N$$

Actual DAC : Mismatches among current sources

$$I_1 = I + \Delta I_1,\ I_2 = I + \Delta I_2,\ I_3 = I + \Delta I_3,\ \ldots,\ I_N = I + \Delta I_N$$

I: Average current

Problem Formulation

Current sources with mismatches

$$I_1 = I + \Delta I_1,\ I_2 = I + \Delta I_2,\ \ldots,\ I_N = I + \Delta I_N$$

Definition of average current

$$I = \frac{1}{N}(I_1 + I_2 + \cdots + I_N) \Rightarrow \Delta I_1 + \Delta I_2 + \cdots + \Delta I_N = 0$$

For digital input = k

$$V_{OUT} = R(I_1 + I_2 + \cdots + I_k) = R\{kI + \underbrace{(\Delta I_1 + \Delta I_2 + \cdots + \Delta I_k)}\}$$

Integral Non-Linearity (INL)

Choose k current sources to INL become small

$$V_{OUT} = R(I_{n1} + I_{n2} + \cdots + I_{nk}) = R\{kI + \underbrace{(\Delta I_{n1} + \Delta I_{n2} + \cdots + \Delta I_{nk})}\}$$

INL should be small

OUTLINE

- Research Background and Objective
- Segmented Current-Steering DAC
- Problem Formulation
- Unit Current Cell Sorting Algorithm
- Simulation Results
- DAC Architecture with Sorting Algorithm
- Conclusion

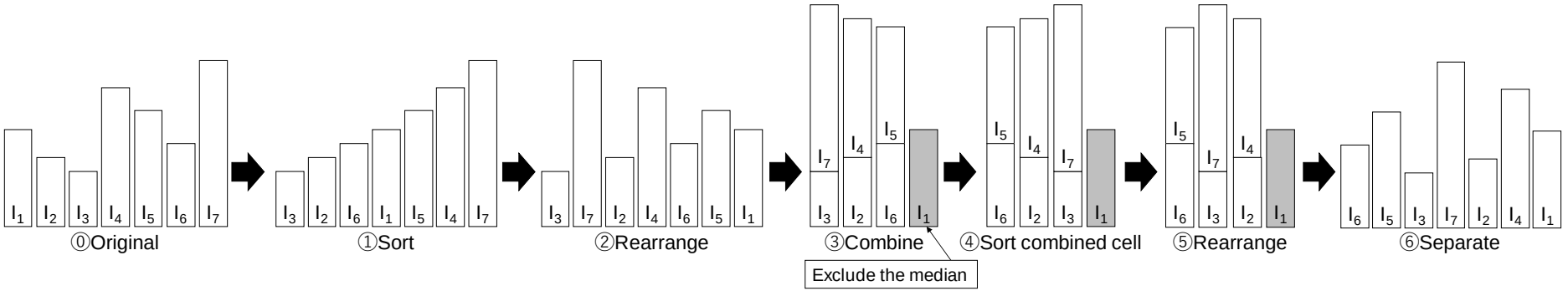
Switching Sequence Post-Adjustment (SSPA)

Features

- Calibration method after fabrication process.
- Change the switching sequence of unit current cells.
- DAC integral linearity is improved.

Concept

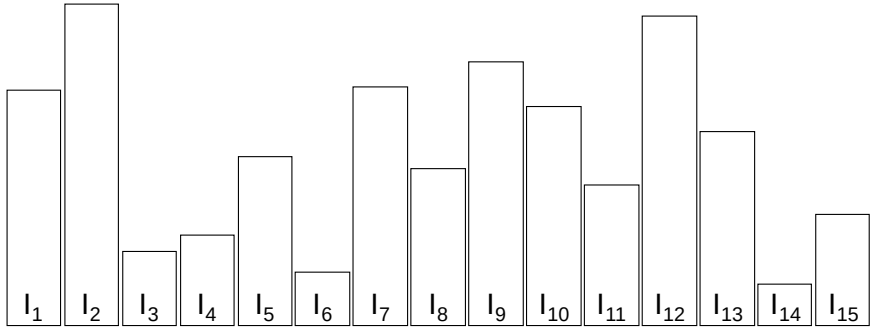
- Offsetting together with larger and smaller mismatch.



T. Chen, G. Gielen (KU Leuven), “A 14-bit 200-MHz Current-Steering DAC with Switching Sequence Post-Adjustment Calibration”, IEEE ASSCC (Nov. 2006)

Proposal “Rearrange”: Step 1

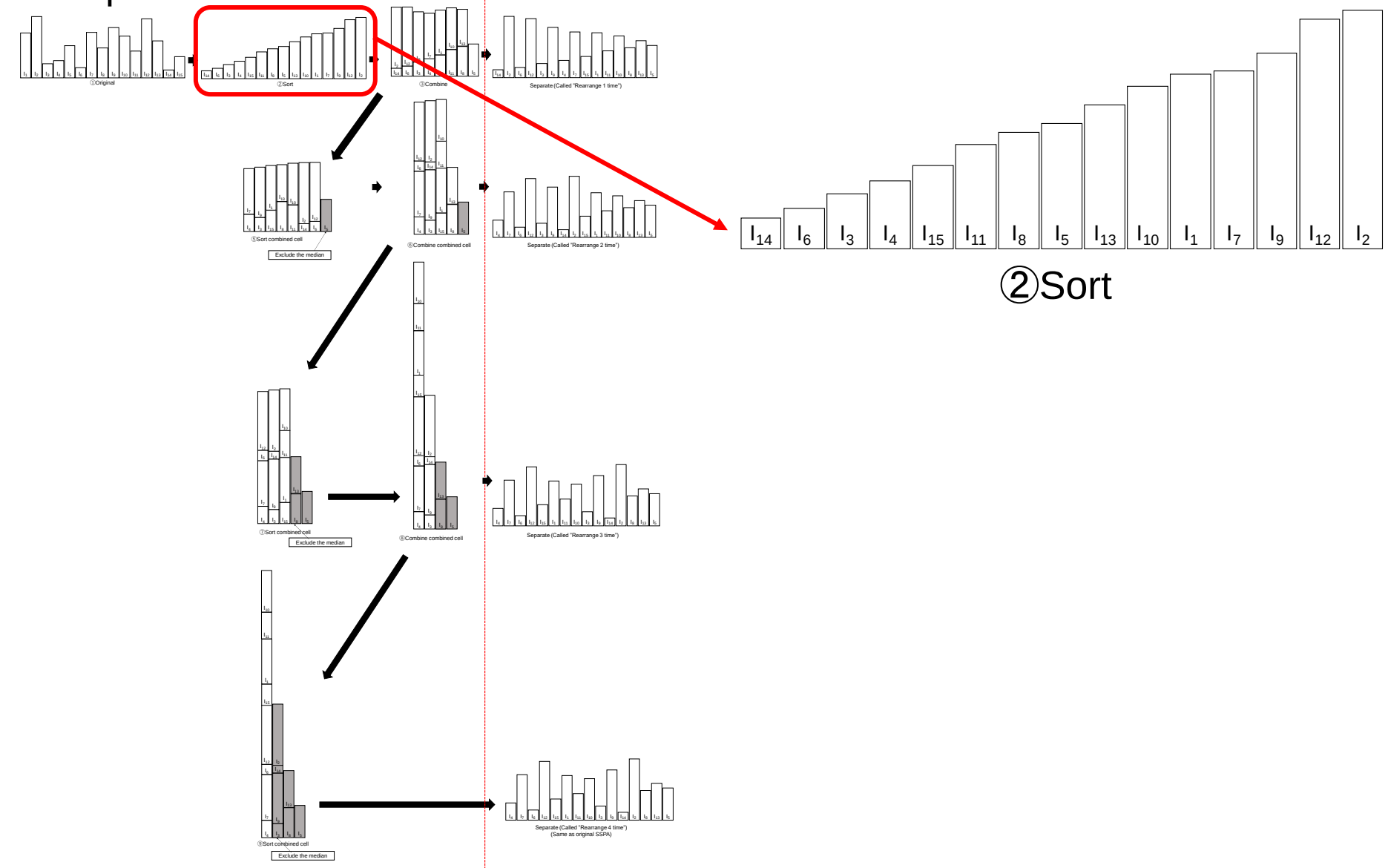
Example: 4bit DAC



①Original current sources order with mismatches

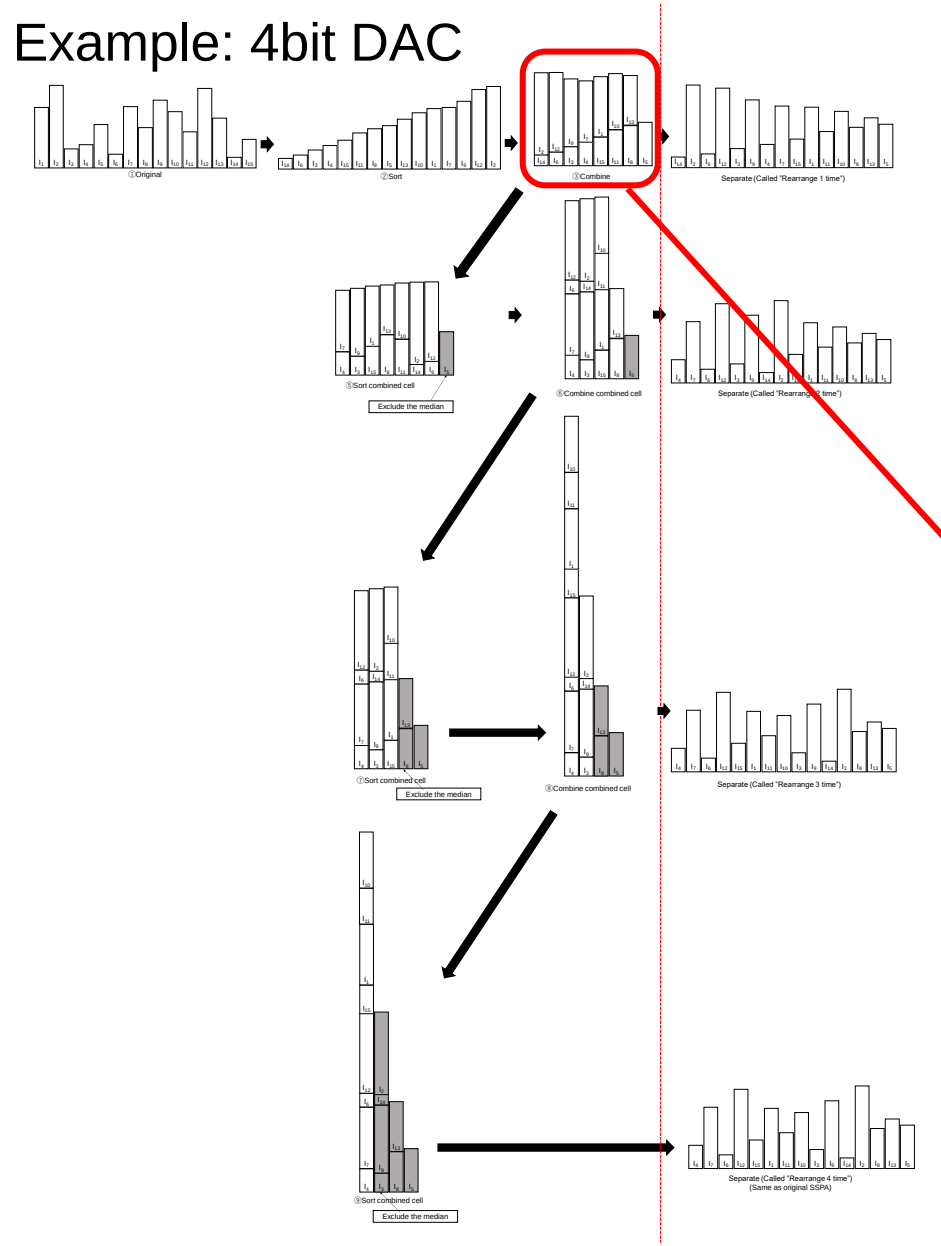
Proposal “Rearrange” : Step 2

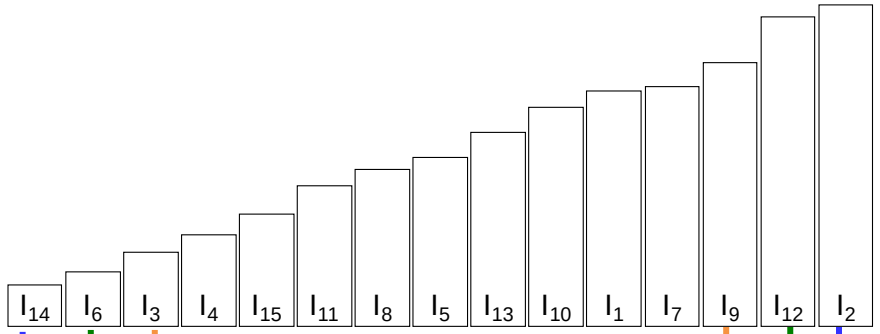
Example: 4bit DAC

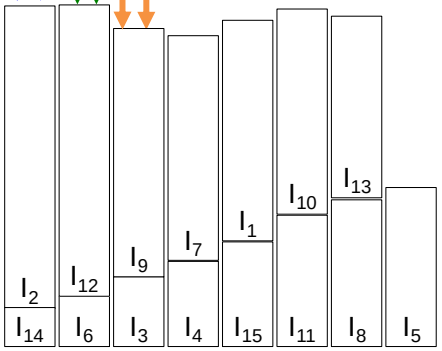


Proposal “Rearrange” : Step 3

Example: 4bit DAC



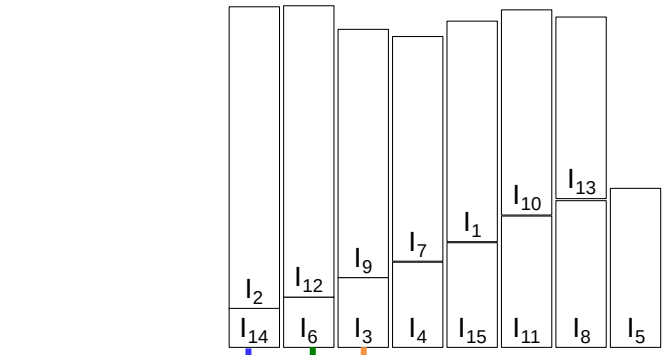




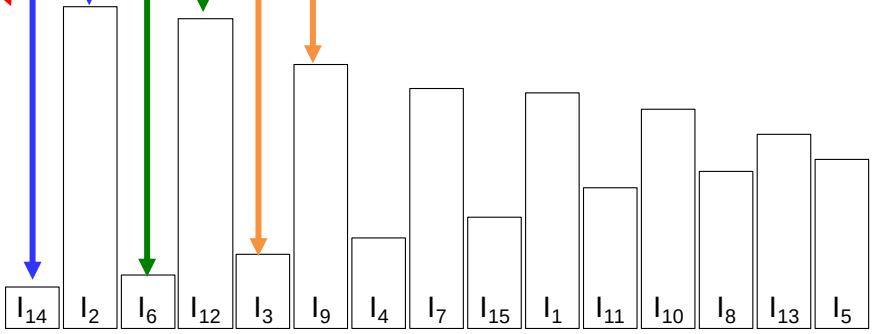
③Combine largest and smallest

Proposal “Rearrange” : Rearrange 1

Example: 4bit DAC



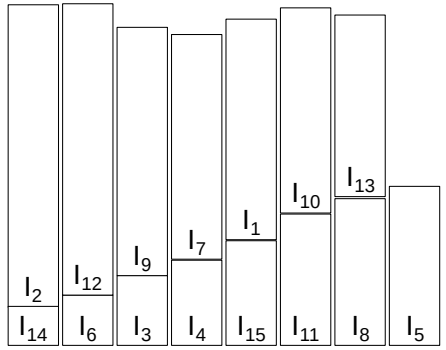
③Combine largest and smallest



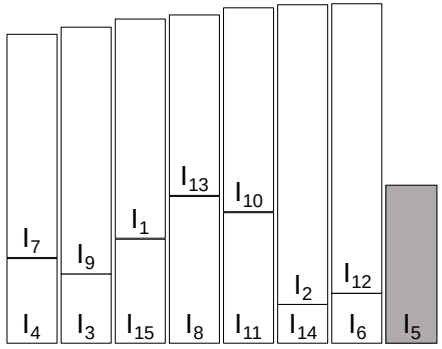
Separate (Called ”Rearrange 1 time”)

Proposal “Rearrange” : Step 4

Example: 4bit DAC



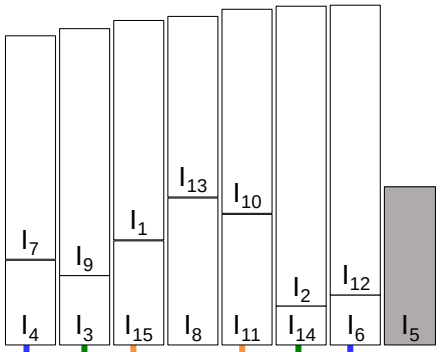
③Combine largest and smallest



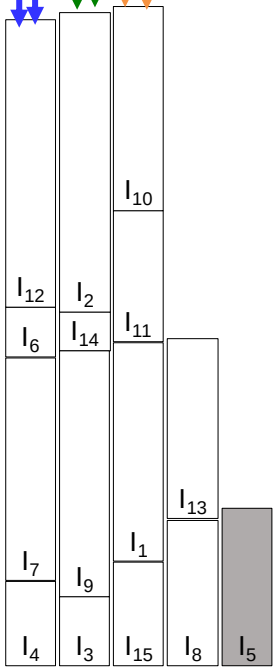
④Sort combined cell
（Exclude the median）

Proposal “Rearrange” : Step 5

Example: 4bit DAC



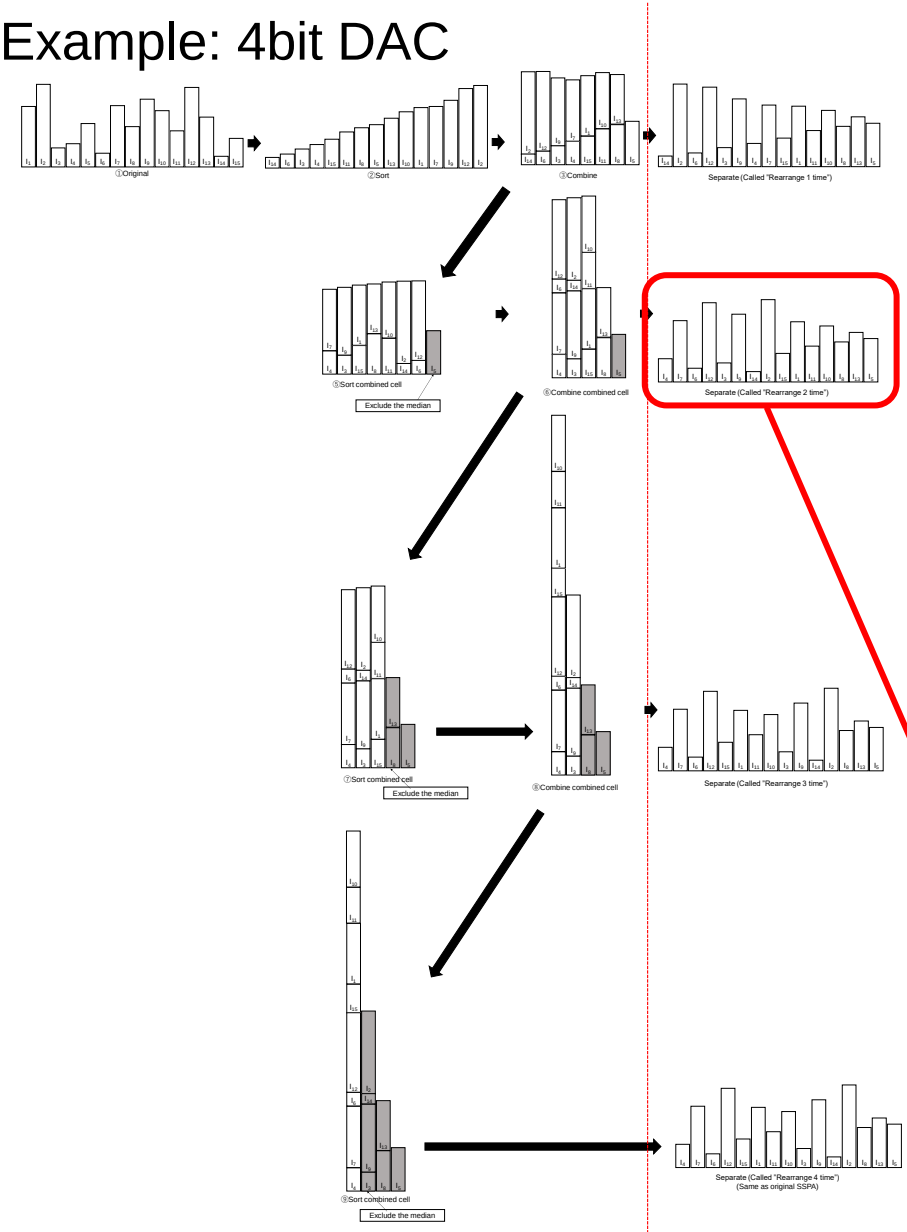
④Sort combined cell

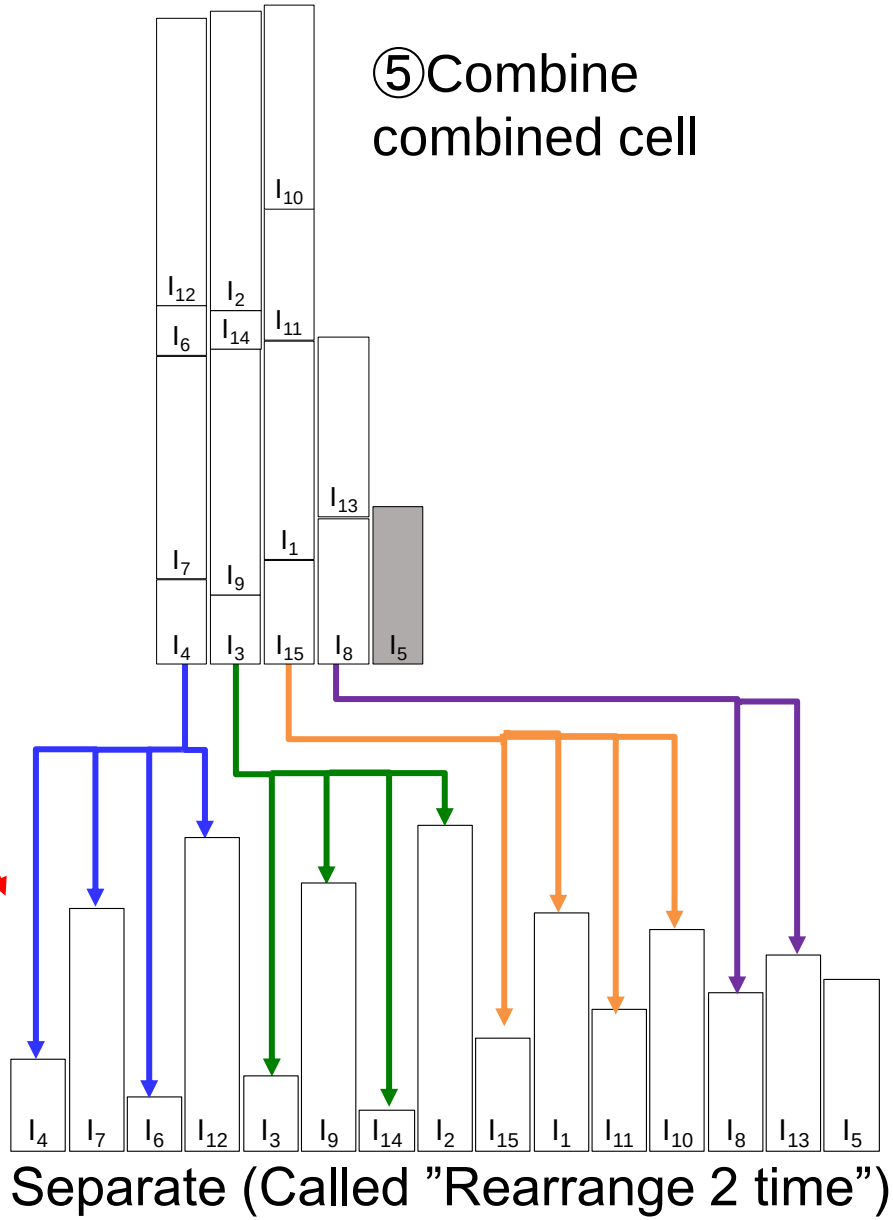


⑤Combine combined cell

Proposal “Rearrange” : Rearrange 2

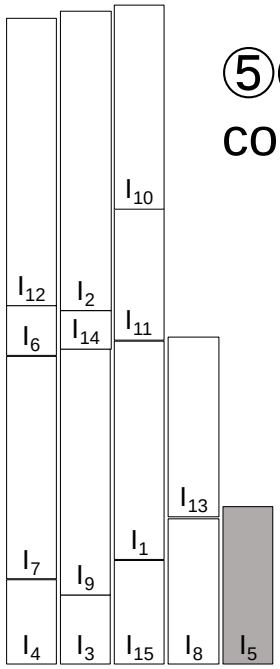
Example: 4bit DAC



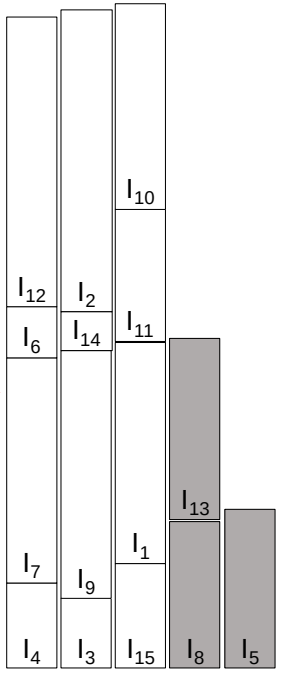


Proposal “Rearrange” : Step 6

Example: 4bit DAC



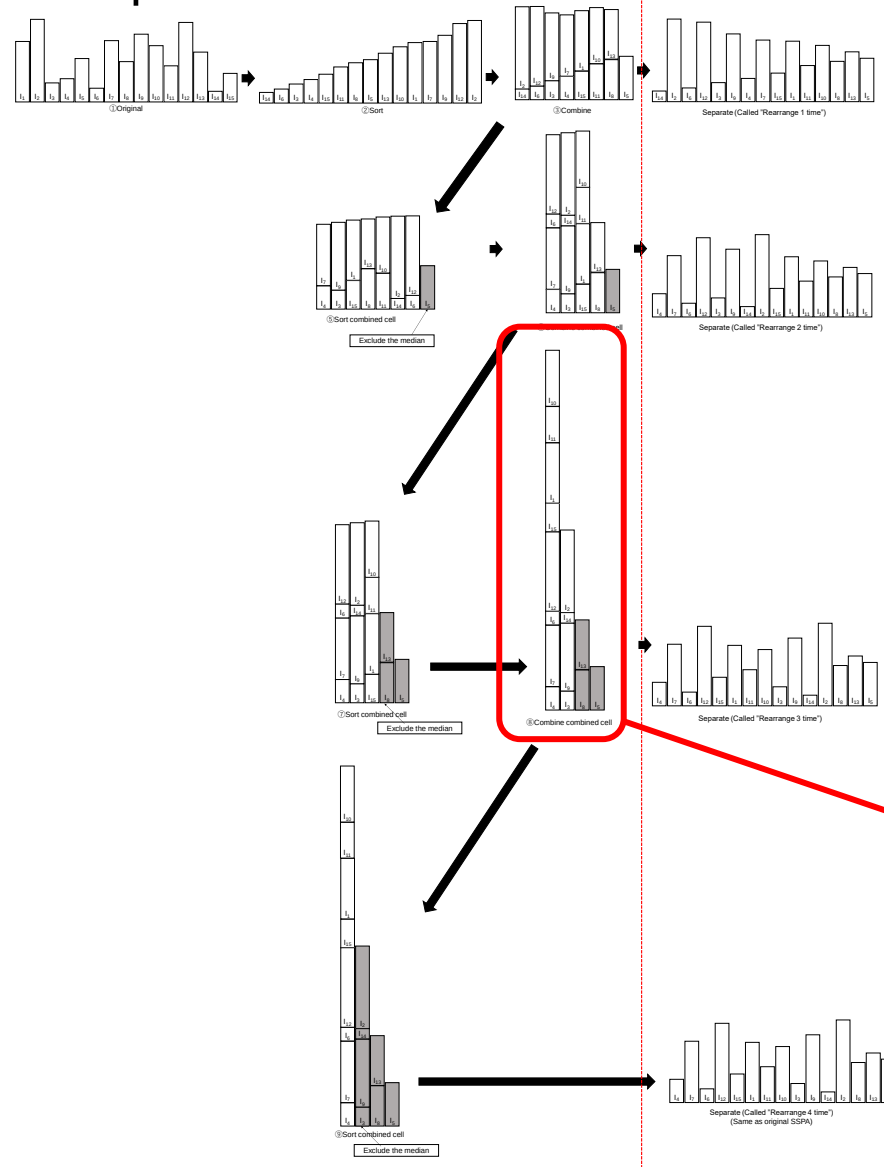
⑤Combine combined cell

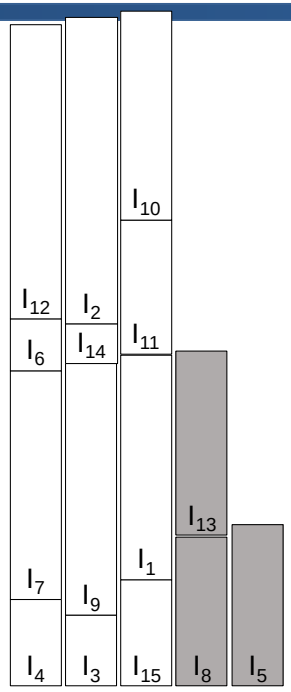


⑥Sort combined cell
（Exclude the median）

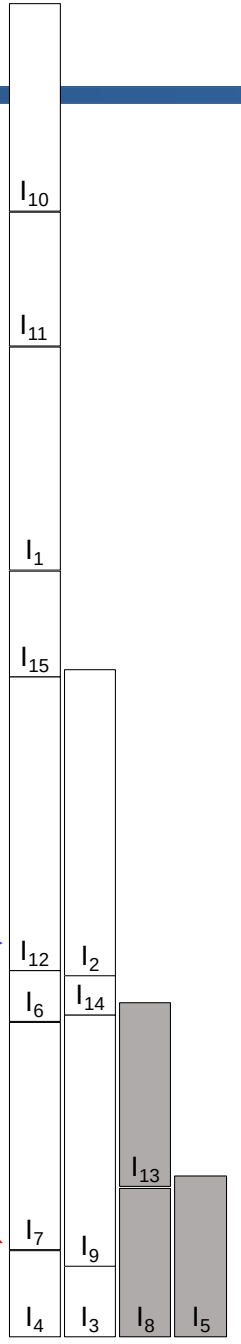
Proposal “Rearrange” : Step 7

Example: 4bit DAC





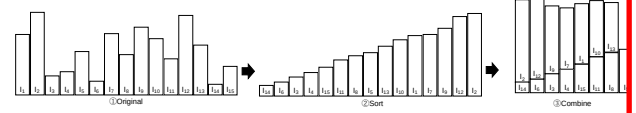
⑥Sort combined cell

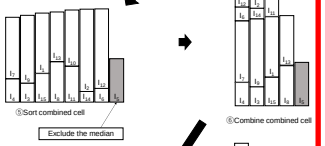


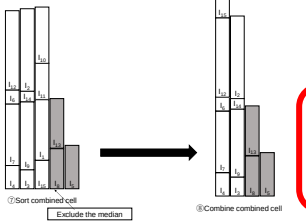
⑦Combine combined cell

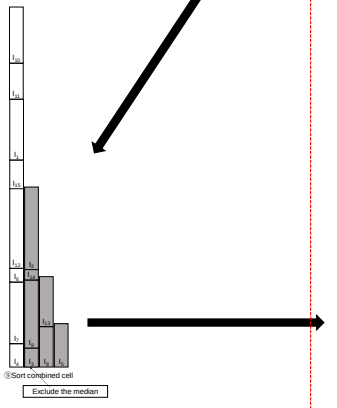
Proposal “Rearrange” : Rearrange 3

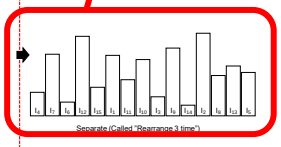
Example: 4bit DAC

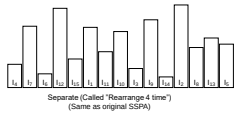




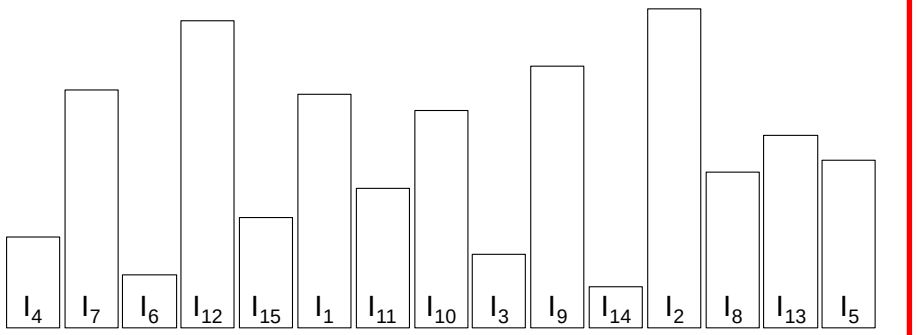


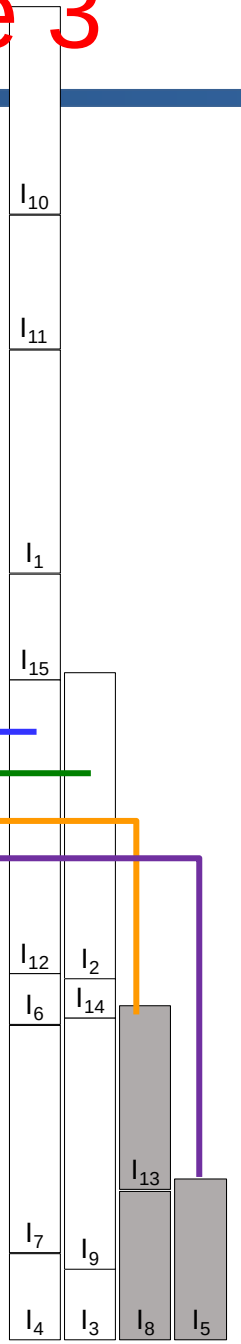






Separate (Called ”Rearrange 3 time”)

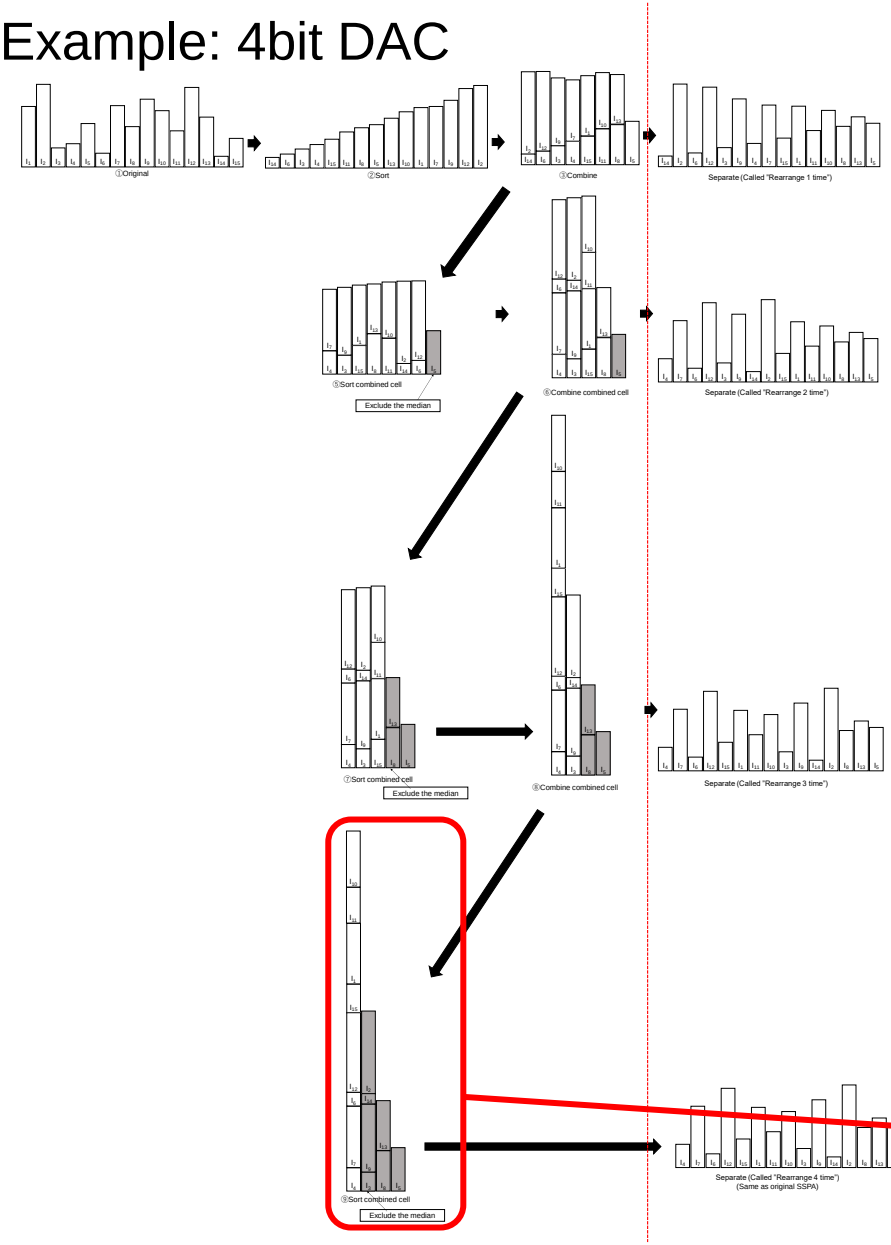


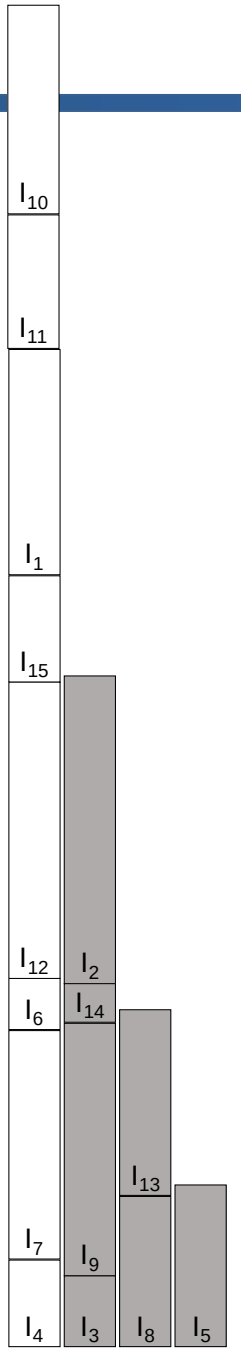


⑦Combine combined cell

Proposal “Rearrange” : Step 8

Example: 4bit DAC

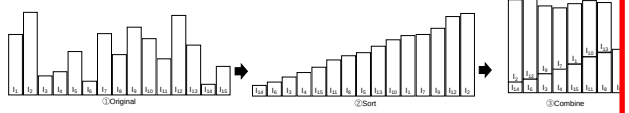


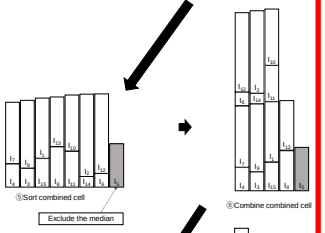


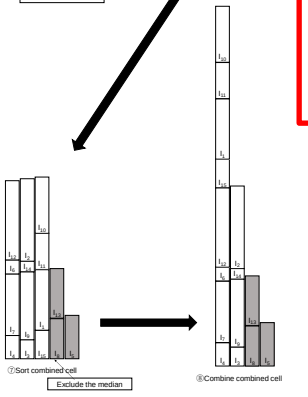
⑧Sort combined cell
（Exclude the median）

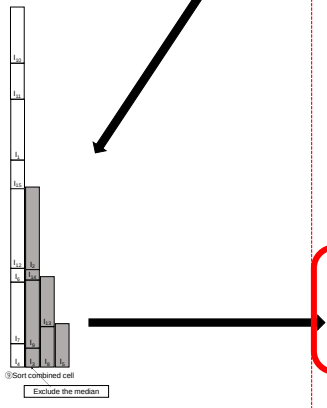
Proposal “Rearrange” : Rearrange 4

Example: 4bit DAC

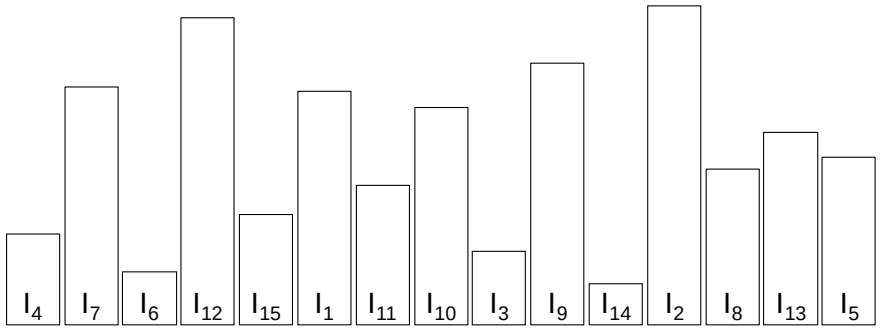


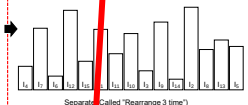


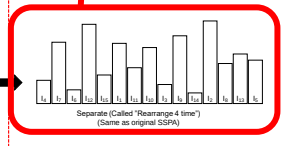


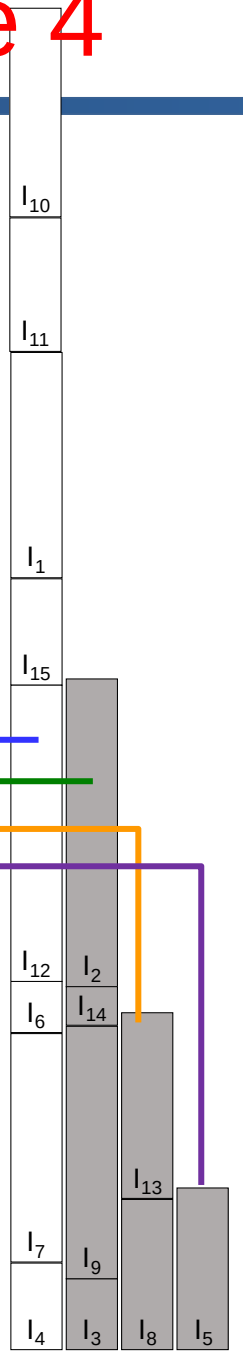


Separate (Called ”Rearrange 4 time”)
(Same as original SSPA)









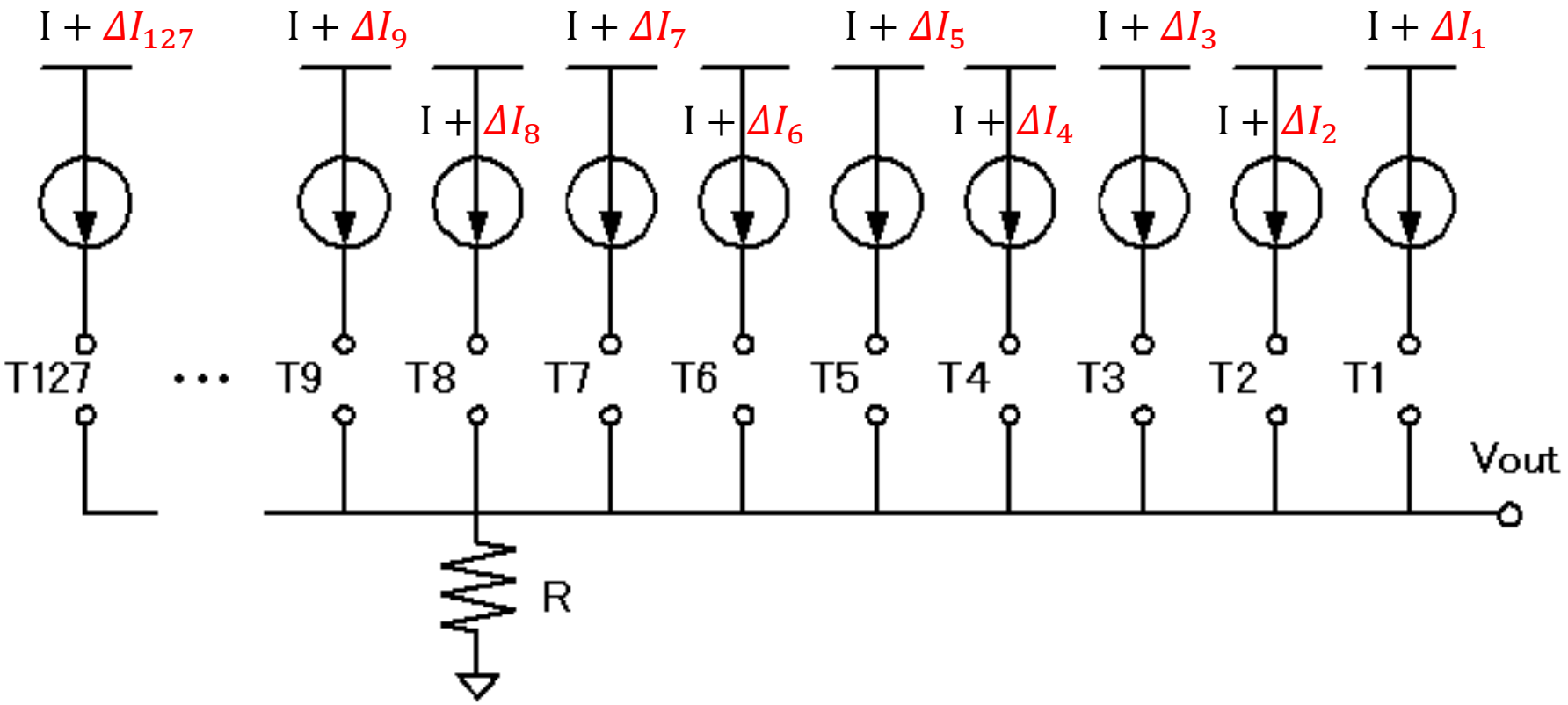
⑧Sort combined cell
（Exclude the median）

OUTLINE

- Research Background and Objective
- Switching Sequence Post Adjustment
- Problem Formulation
- Unit Current Cell Sorting Algorithm
- Simulation Results
- DAC Architecture with Sorting Algorithm
- Conclusion

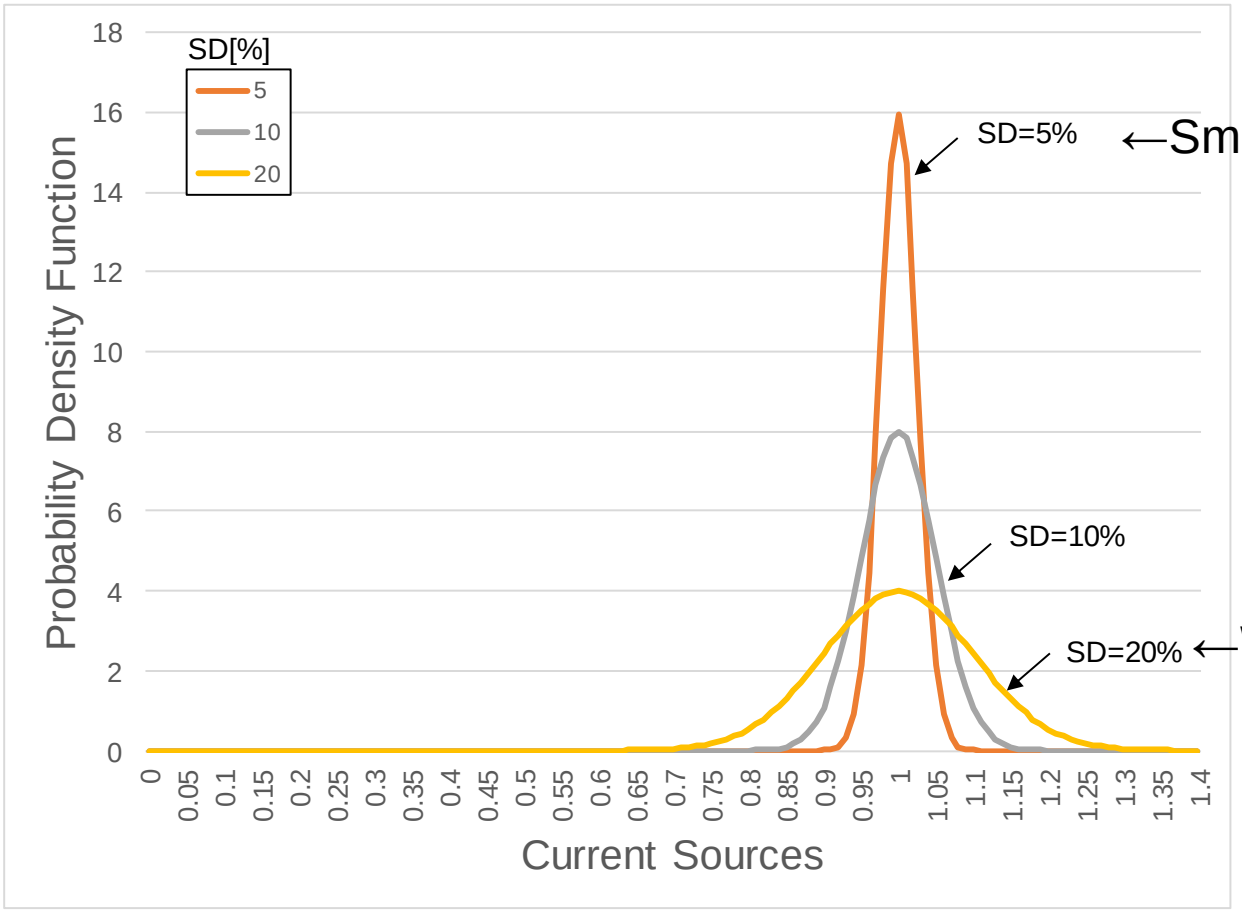
Simulation Conditions

Assuming a 7-bit DAC, 127 current sources are used.
The current sources have mismatches. (ΔI_i)



Mismatch variation as a parameter

Mismatches vary within the range of SD [%].
The simulation was performed by varying the degree of mismatches from 1% to 20% as parameter SD.



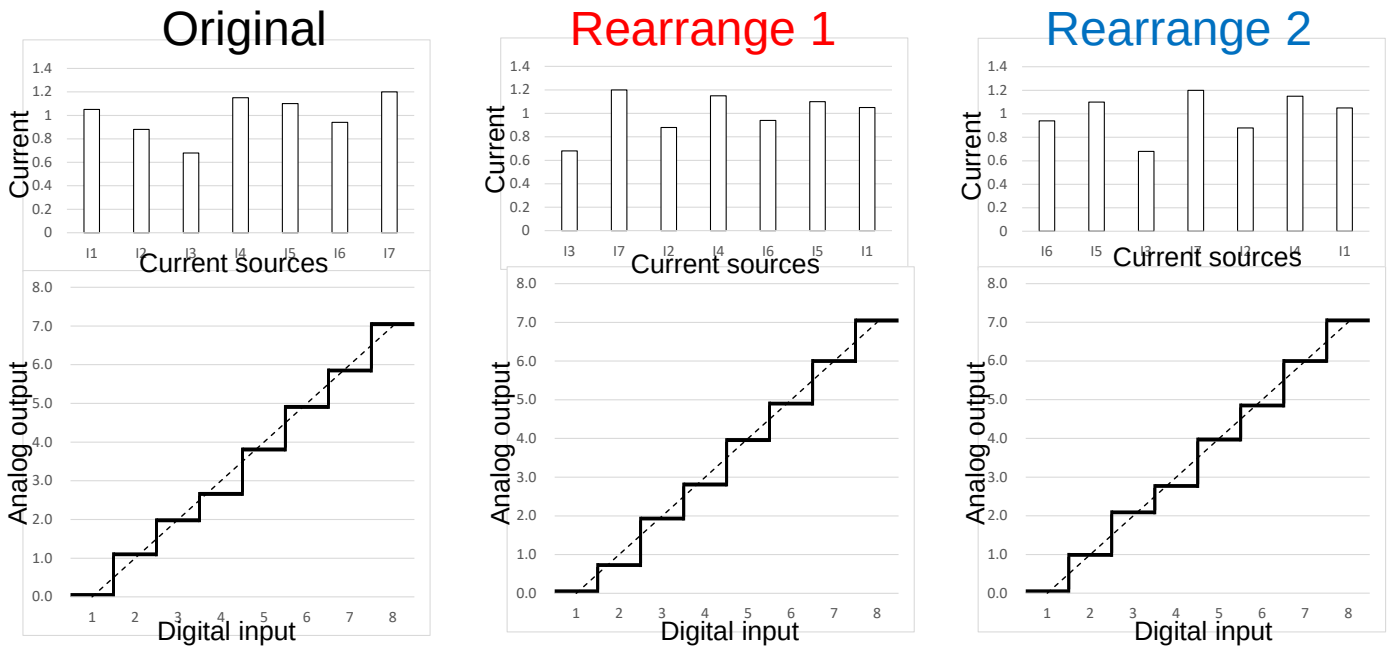
Example of the current sources mismatches

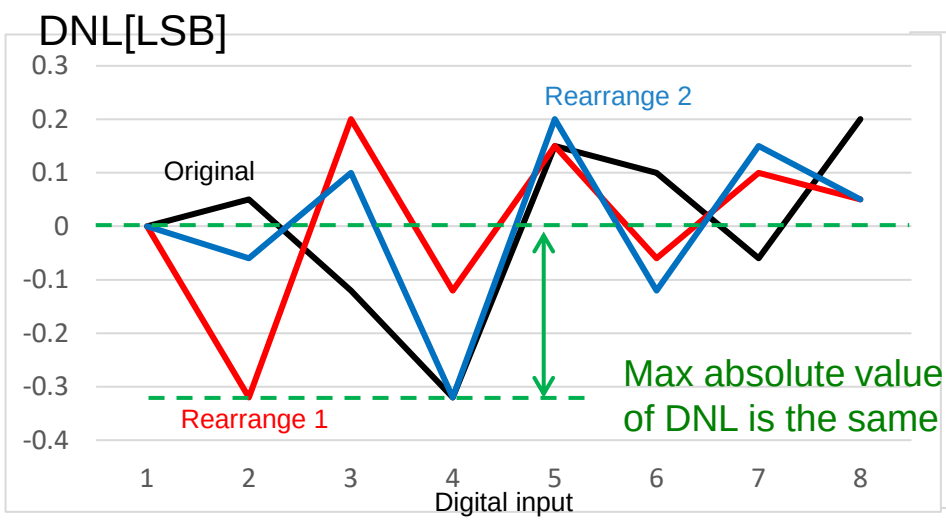
DNL & INL of Rearranges

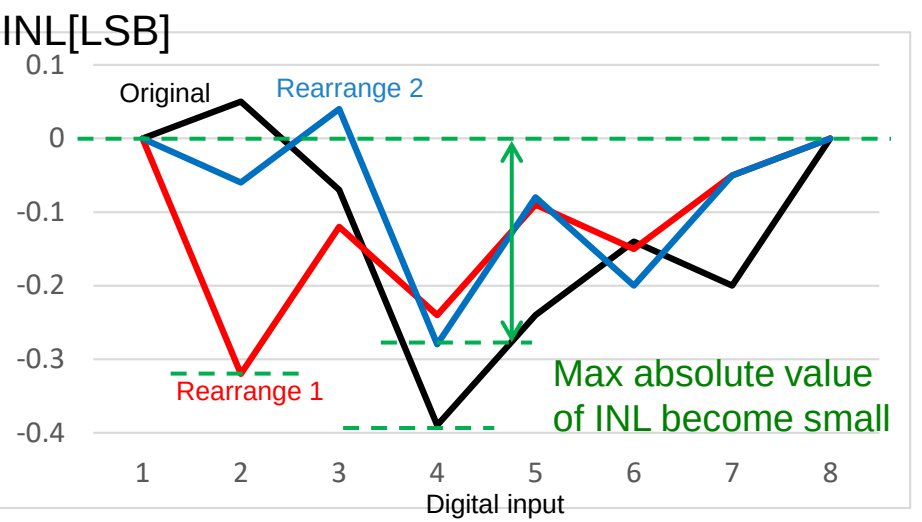
Example: 3bit DAC

Current sources with mismatches

Digital input & Analog output

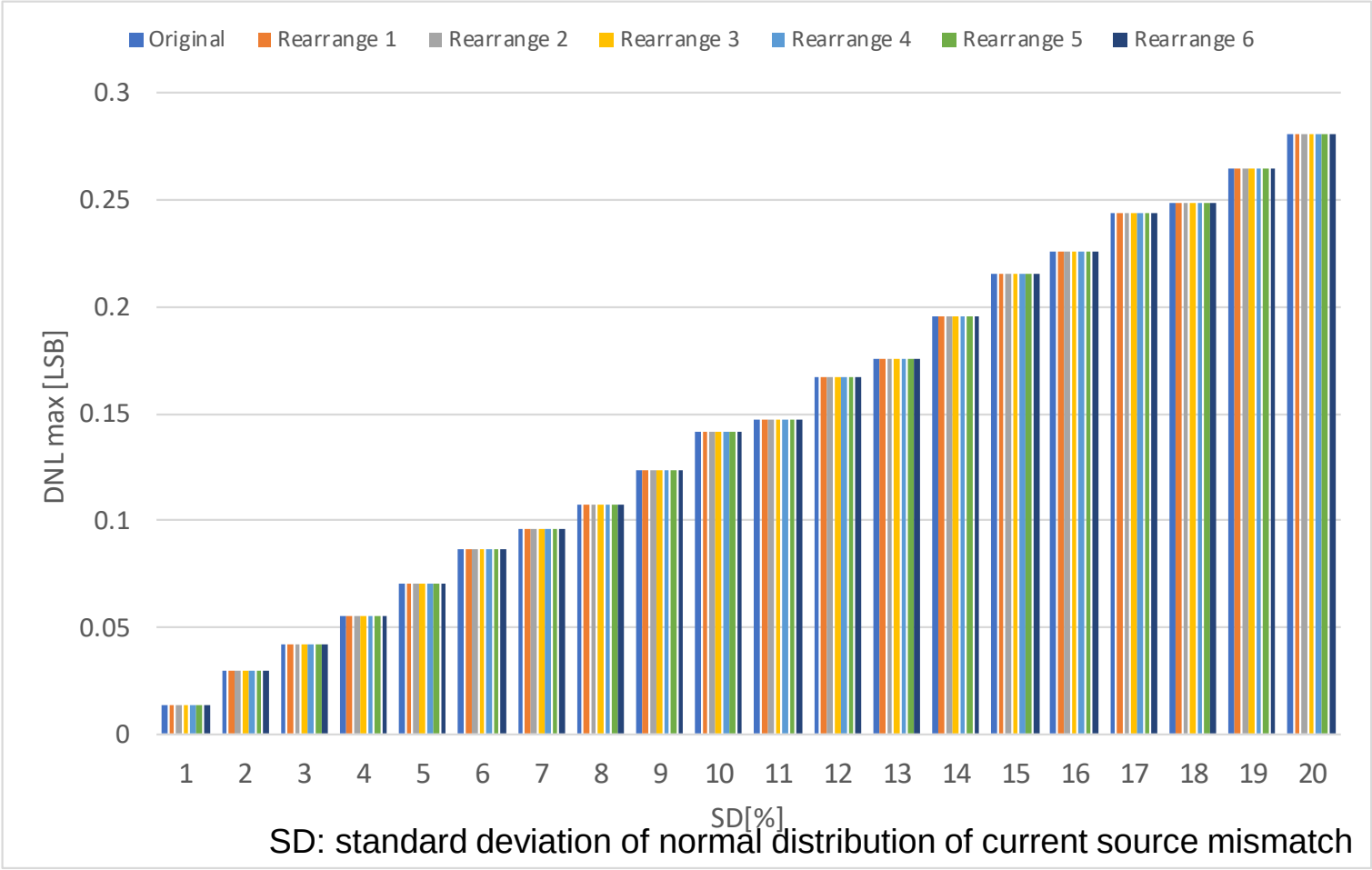






DNLmax

DNLmax=maximum(DNL_1 : DNL_{127})

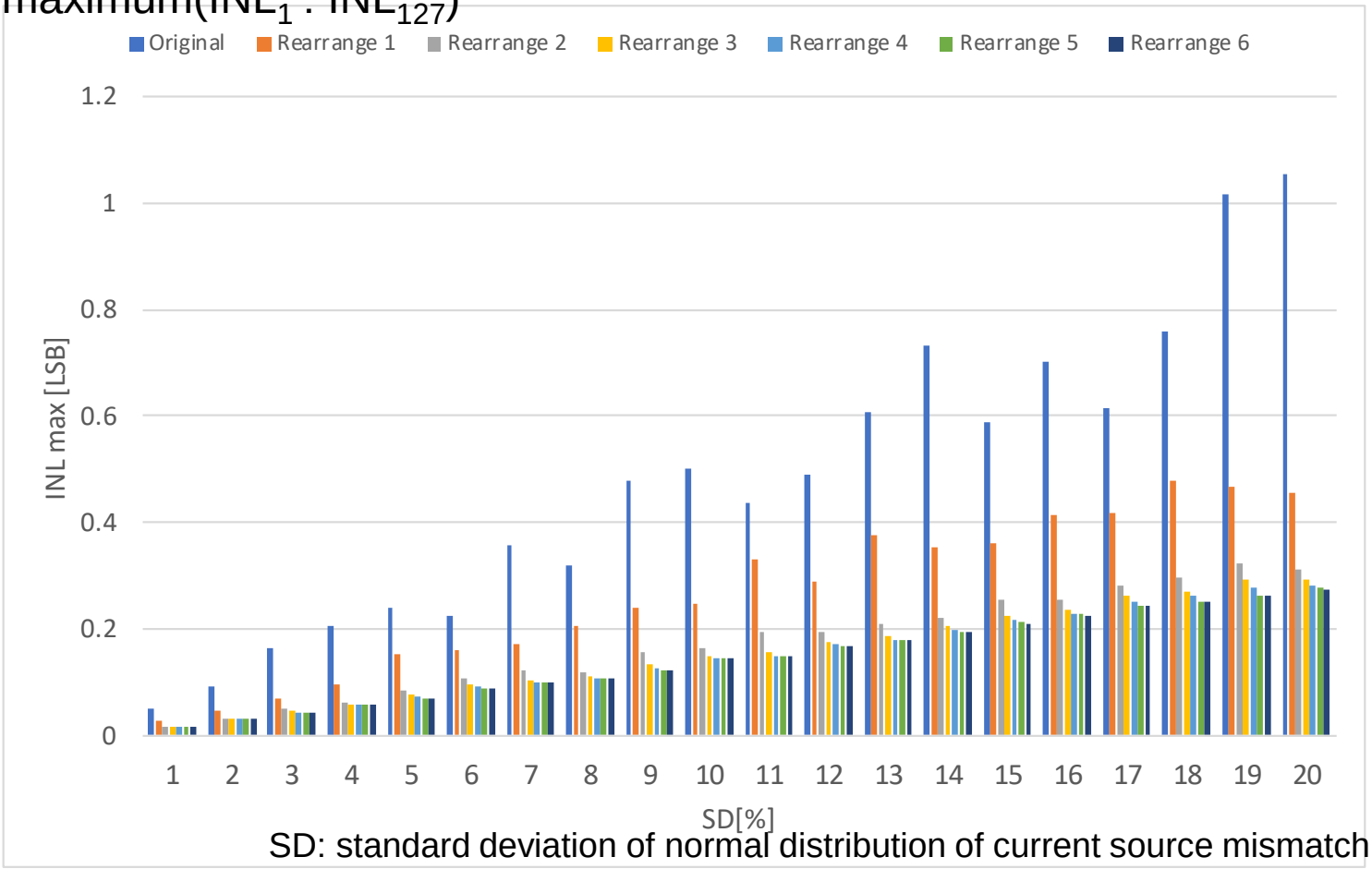


SD: standard deviation of normal distribution of current source mismatch

- Rearranges does not change DNL.
- DNL is proportional to SD(standard deviation of current source mismatch).

INLmax

INLmax=maximum(INL_1 : INL_{127})



SD: standard deviation of normal distribution of current source mismatch

- Rearranges reduce INL (average of reduction ratios are…)
  - Rearrange 1 : 56%
  - Rearrange 2 : 36%
  - Rearrange 3 : 32%
  - Rearrange 4 : 31%
  - Rearrange 5 : 31%
  - Rearrange 6 : 31%

➡ Rearrange 2 is enough

OUTLINE

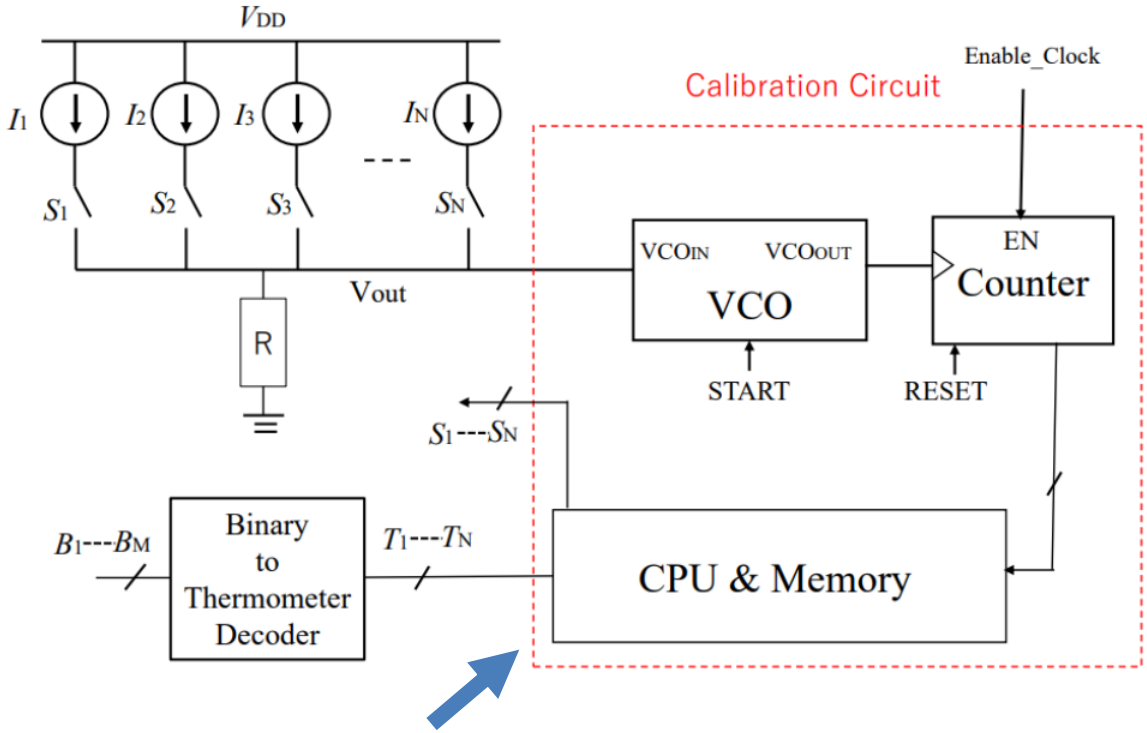
- Research Background and Objective
- Segmented Current-Steering DAC
- Problem Formulation
- Unit Current Cell Sorting Algorithm
- Simulation Results
- DAC Architecture with Sorting Algorithm
- Conclusion

DAC Architecture with Calibration using Sorting Algorithm

- Current comparator is used in conventional SSPA.

T. Chen, G. Gielen (KU Leuven), “A 14-bit 200-MHz Current-Steering DAC with Switching Sequence Post-Adjustment Calibration”, IEEE ASSCC (Nov. 2006)

- Our proposal is VCO instead of comparator.

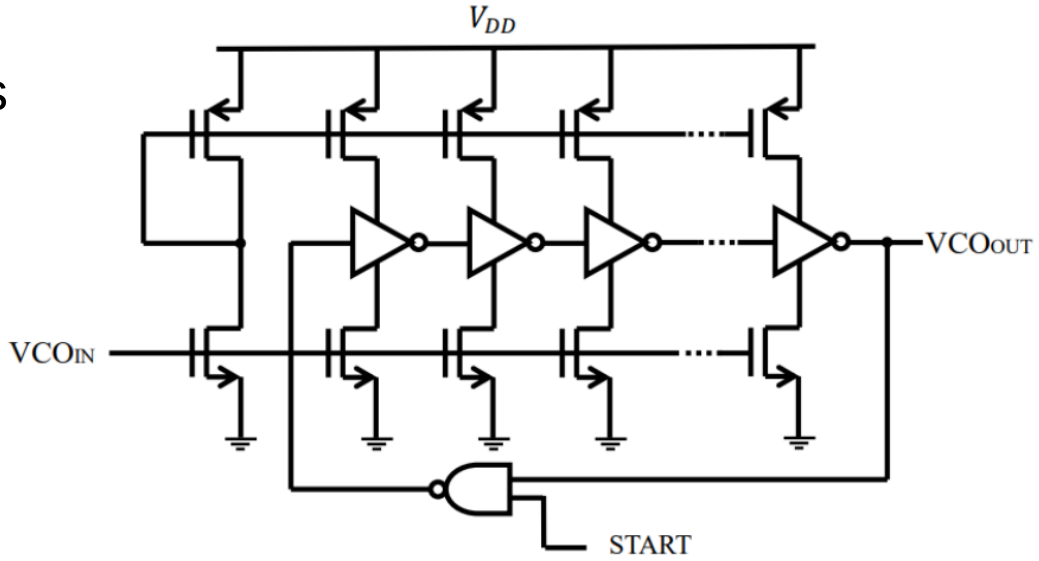


- Implementation with only digital circuit

  No need for a current comparator

VCO with Current-Controlled Inverters and START Circuit

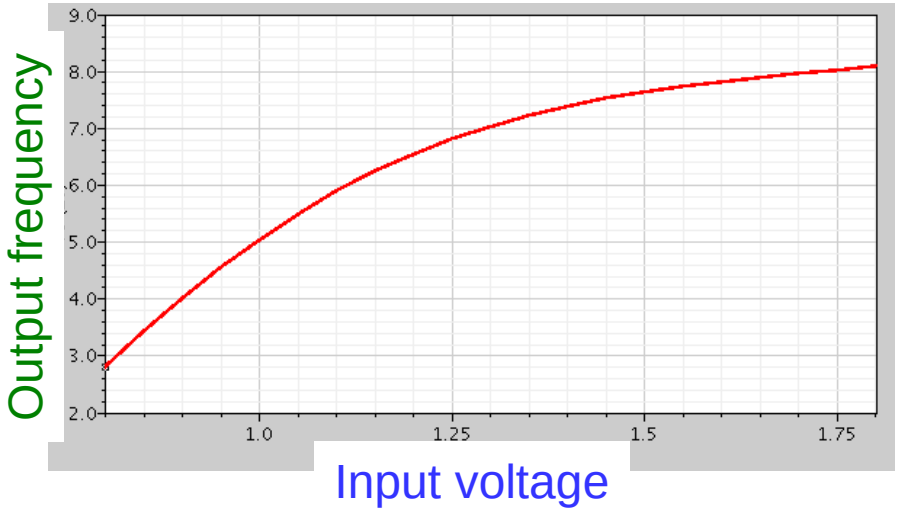
Ring oscillators



Input voltage

Output frequency

SPICE simulation results



Nonlinear, but monotonic characteristics

OUTLINE

- Research Background and Objective
- Segmented Current-Steering DAC
- Problem Formulation
- Unit Current Cell Sorting Algorithm
- Simulation Results
- DAC Architecture with Sorting Algorithm
- Conclusion

Conclusion

- Investigated the segmented current-steering DAC linearity improvement algorithm using unit current cell sorting algorithm (SSPA).

- Proposed "Rearrange" stops the SSPA step in the middle.

- Simulation results show:
  - "Rearrange2" reduces INL enough.
    (conventional SSPA is same "Rearrange6")
  - DNL is proportional to variation size of current source mismatch

- Proposed a digital implementation method of the SSPA.

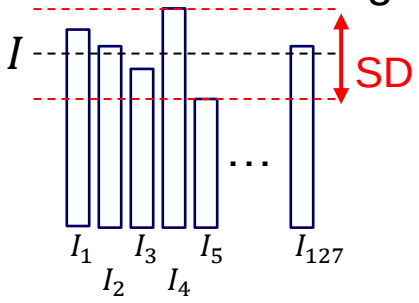
Thank you for listening

Kobayashi Laboratory

群馬大学 GUNMA UNIVERSITY

Simulation Procedure

ΔI_i are generated by rand function in C language.
Varies within the range of SD

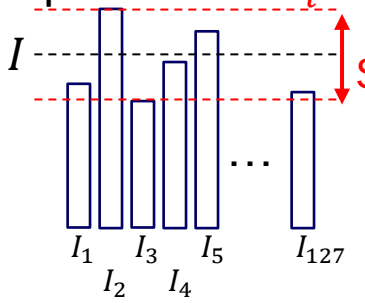


Original
- DNLmax
- INLmax

Rearrange 1
- DNLmax
- INLmax

… Rearrange 7
- DNLmax
- INLmax

Other pattern of ΔI_i



Original
- DNLmax
- INLmax

Rearrange 1
- DNLmax
- INLmax

… Rearrange 7
- DNLmax
- INLmax

⋮

10 patterns

Original
- average of DNLmax
- average of INLmax

Rearrange 1
- average of DNLmax
- average of INLmax

…

Q&A

●質問：
　このキャリブレーションでDACのスピードはおそくならないのか？

●答え：
最初に1回キャリブレーションをやればよいので、その後は遅くならない。
（毎サイクルごとにキャリブレーションをやるわけではない。）