

2016 International Symposium on

**VLSI Design, Automation and Test**



# **DAC Linearity Improvement Algorithm With Unit Cell Sorting Based on Magic Square**

**Masashi Higashino**

**Shaiful Nizam Mohyar,** Haruo Kobayashi

**Division of Electronics and Informatics**

**Gunma University, Japan**

**Universiti Malaysia Perlis, Malaysia**

# OUTLINE

- Research Objective
- Current Steering DAC
- What is Magic Square ?
- Proposed Algorithm
- Simulation Results
- Conclusion

# OUTLINE

- Research Objective
- Current Steering DAC
- What is Magic Square ?
- Proposed Algorithm
- Simulation Results
- Conclusion

# Research Objective

## Research Background

- Demand for DAC in communication systems
  - High linearity
  - High spurious free dynamic range (SFDR)

## Our Approach

- Unary DAC linearity improvement
  - Unit cell sorting algorithm
  - Based on **Magic Square**
  - Digital method
  - No analog part modification

New!!!

# OUTLINE

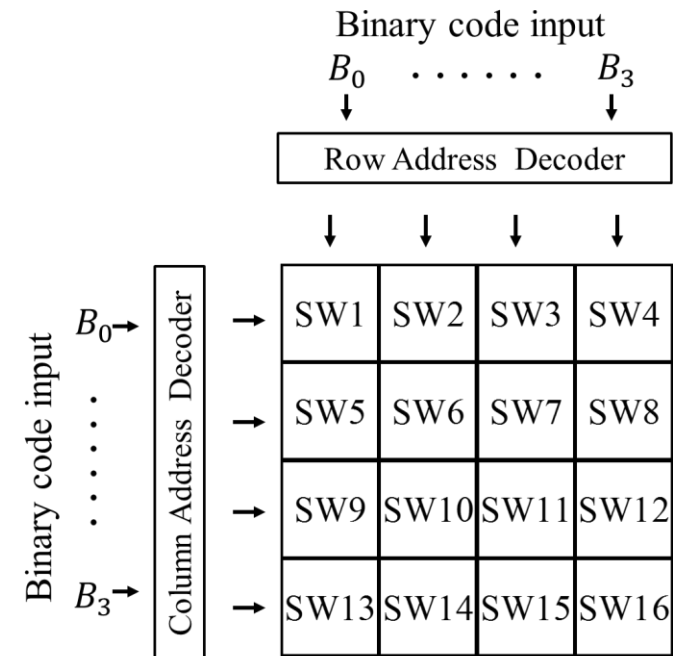
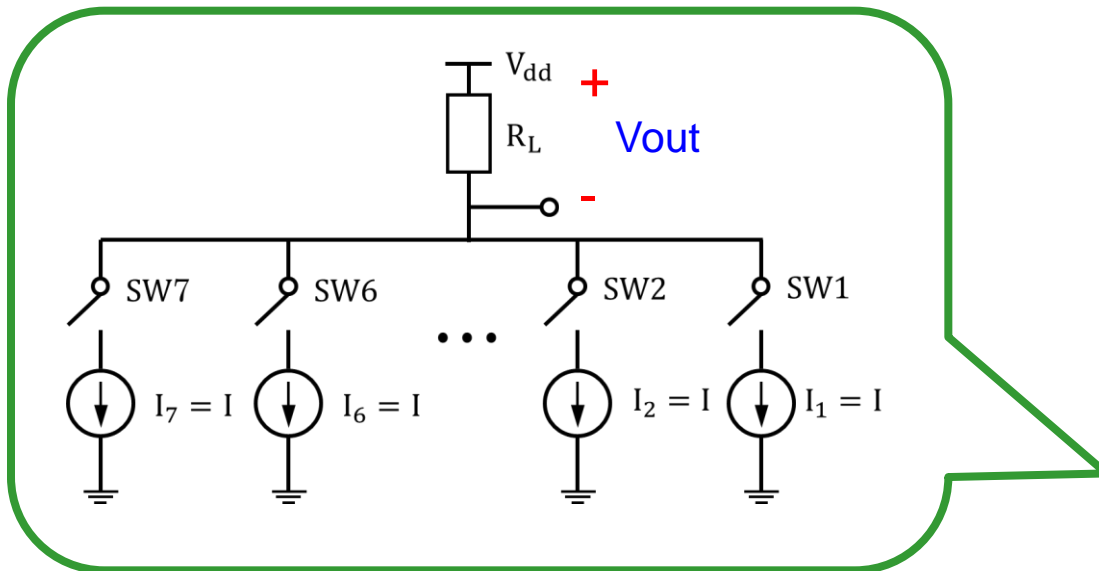
- Research Objective
- **Current Steering DAC**
- What is Magic Square ?
- Proposed Algorithm
- Simulation Results
- Conclusion

# Circuit and Features of Unary Current-Steering DAC

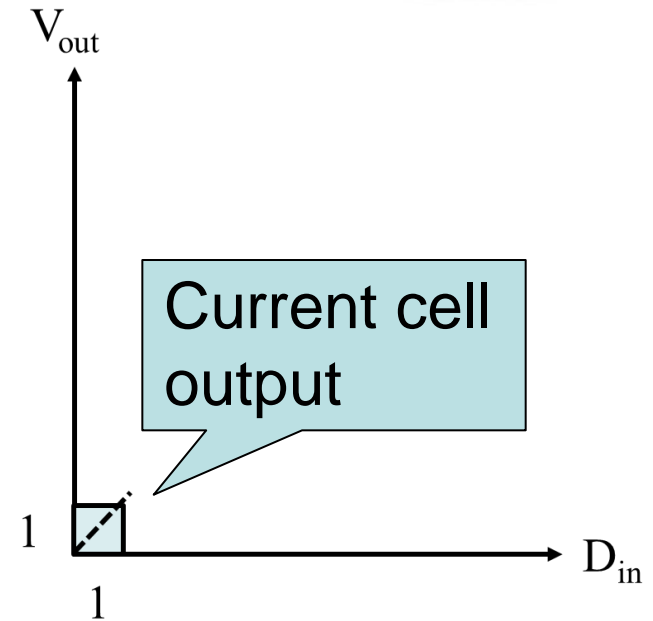
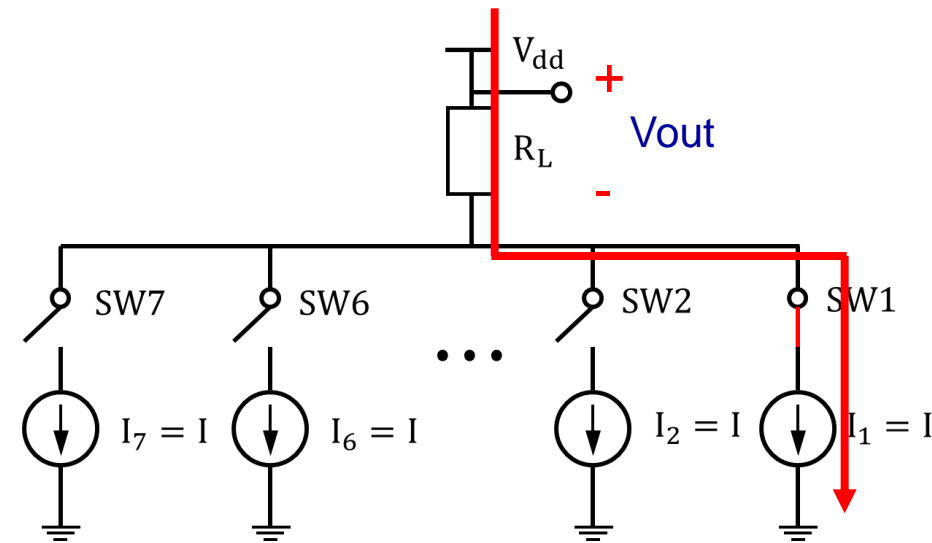
- Identical current sources
- Small glitch
- Inherent monotonicity
- High speed



- Large circuits
  - Decoder
  - Many switches and current sources



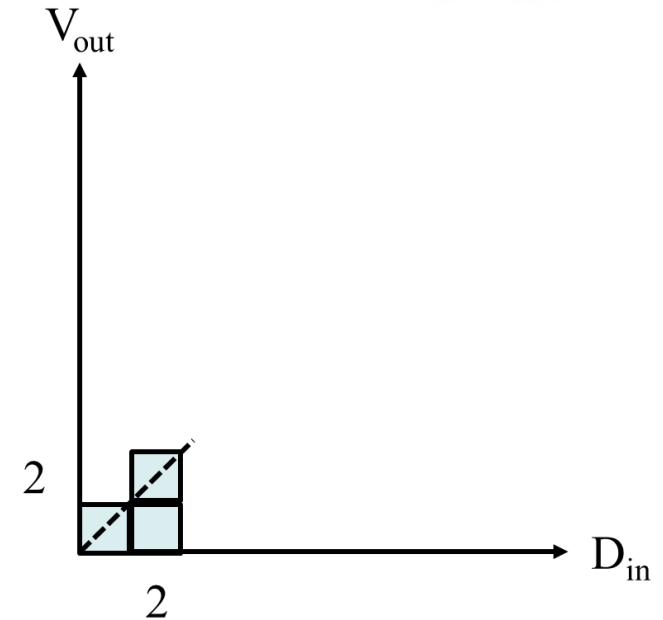
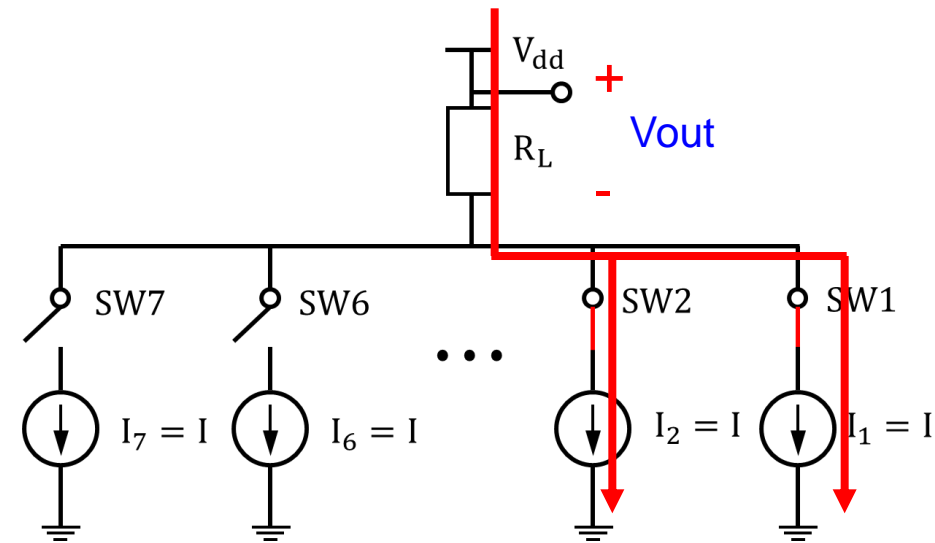
# Operation of Current Steering DAC (1)



Digital input = 1 , 1 current source.

$$V_{out} = R_L I_1$$

# Operation of Current Steering DAC (2)

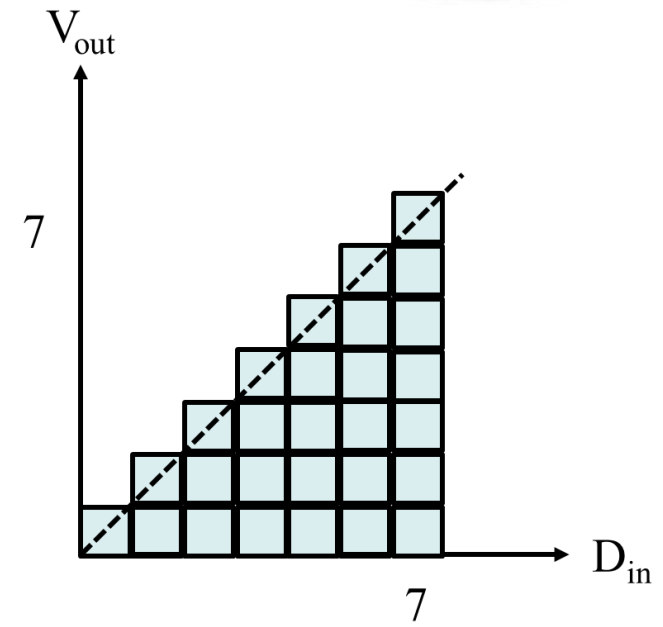
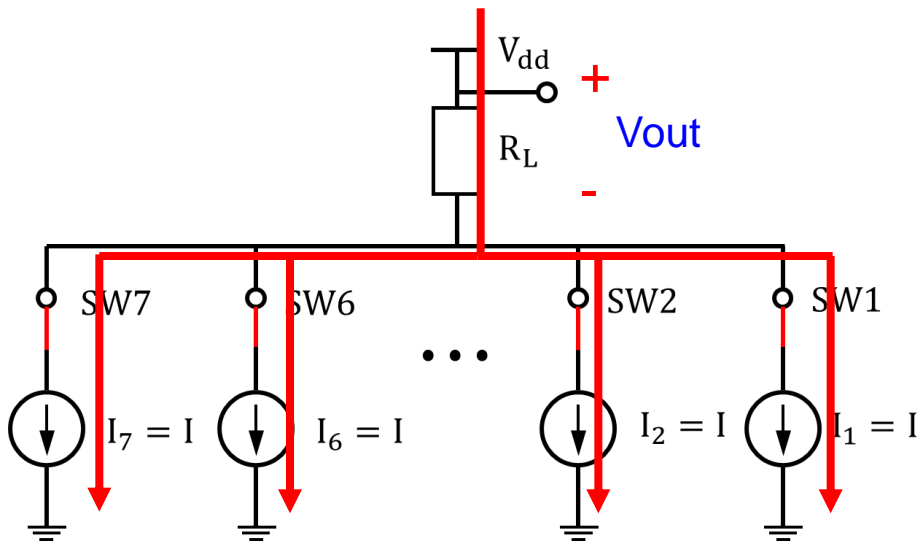


Digital input = 1 , 1 current source.  
 “ = 2 , 2 current sources.

$$V_{out} = R_L(I_1 + I_2)$$



# Operation of Current Steering DAC (3)



Digital input = 1 , 1 current source.  
 “ = 2 , 2 current sources.  
 !  
 “ = 7 , 7 current sources.

$$V_{out} = R_L(I_1 + I_2 + \dots + I_7)$$

$$I_1 = I_2 = \dots = I_6 = I_7$$

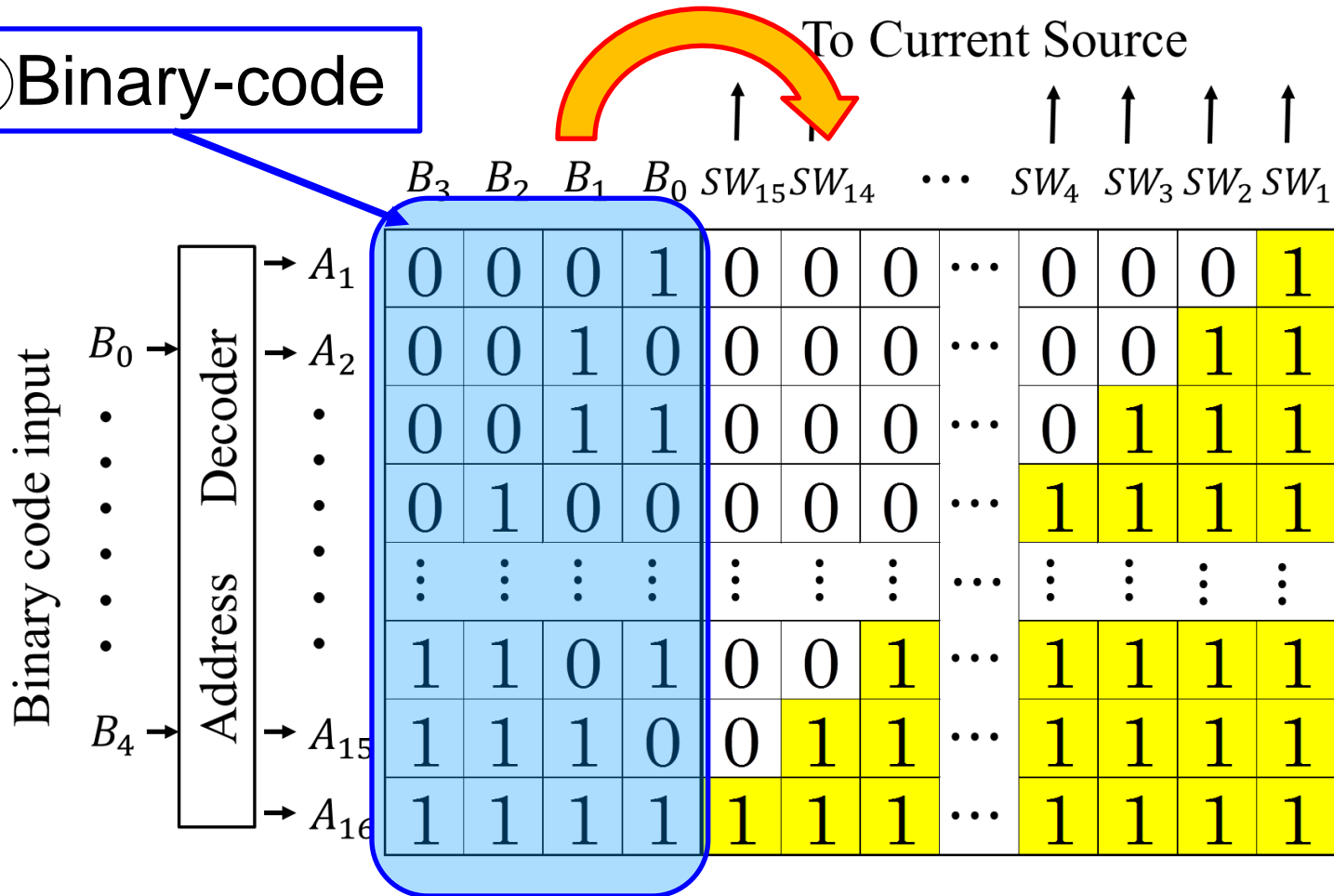


DAC is perfectly linear

# Conventional Unary DAC Decoder

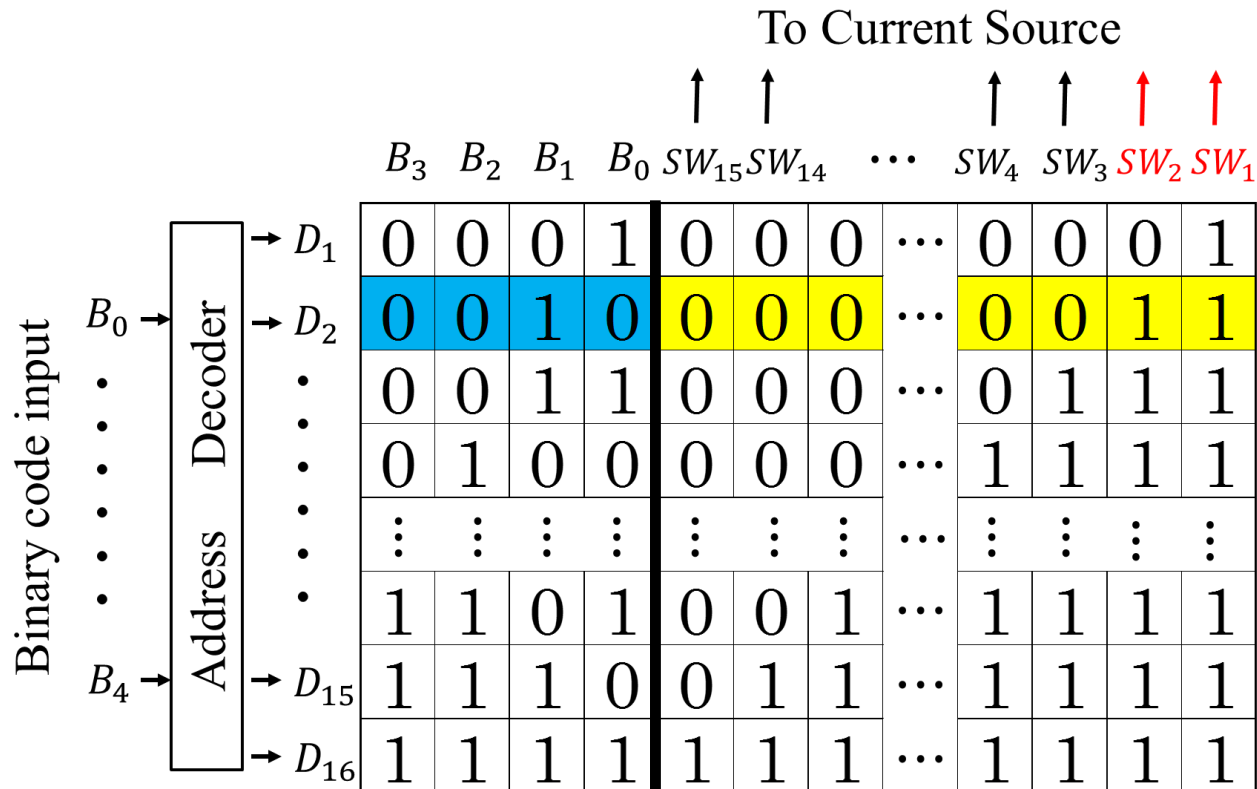
## ② Thermometer-code

## ① Binary-code



## Example 1

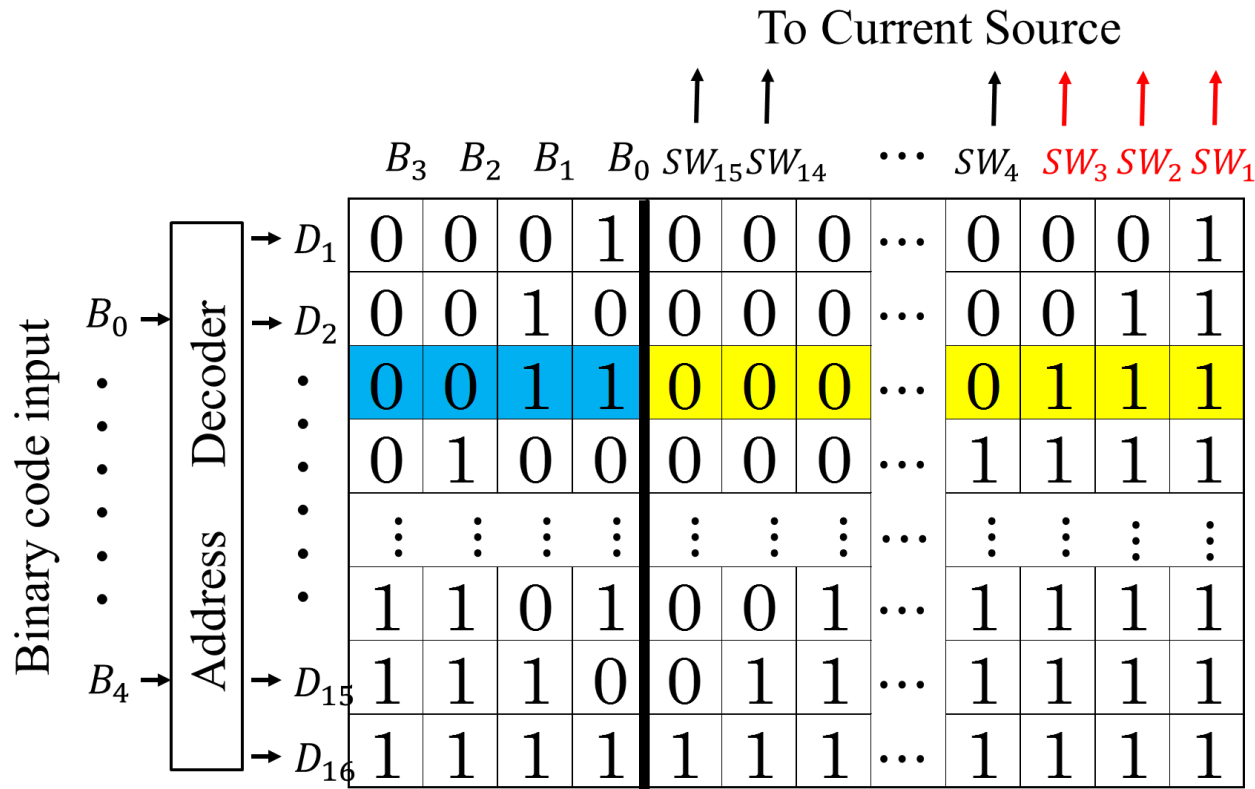
- Digital binary input (0010)
- ➔ Thermometer code (0000 0000 0000 0011)
- ➔ 2 current cells turn on.



# Operation of Unary DAC Decoder

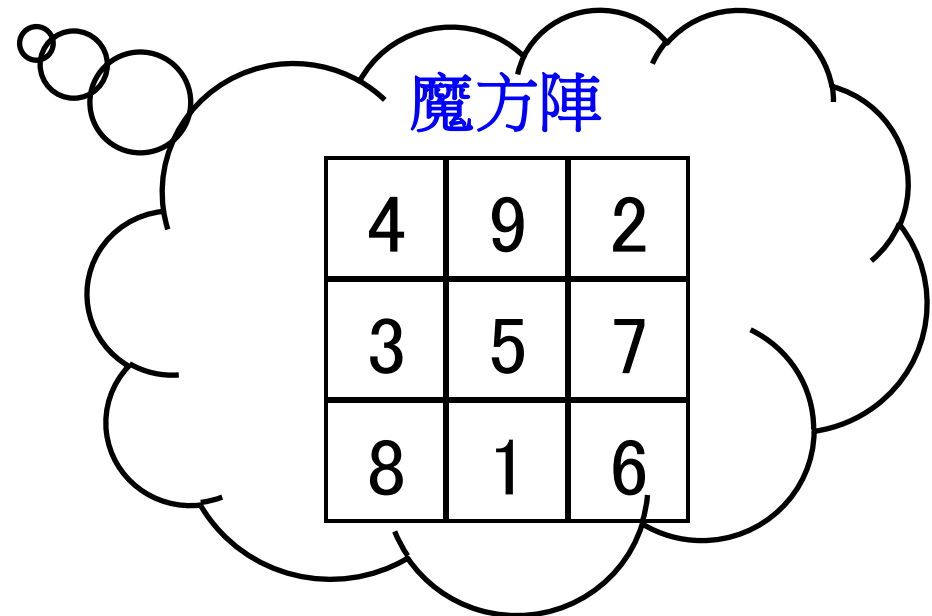
## Example 2

- Digital binary input (00**11**)
- ➔ Thermometer code (0000 0000 0000 0**111**)
- ➔ 3 current cells turn on.



# OUTLINE

- Research Objective
- Current Steering DAC
- What is Magic Square ?
- Proposed Algorithm
- Simulation Results
- Conclusion



# What is Magic Square ?

- Classical mathematics
- Origin from Chinese academia
- “Constant sum” characteristics
- Varieties of magic squares

**河 圖 洛 書**

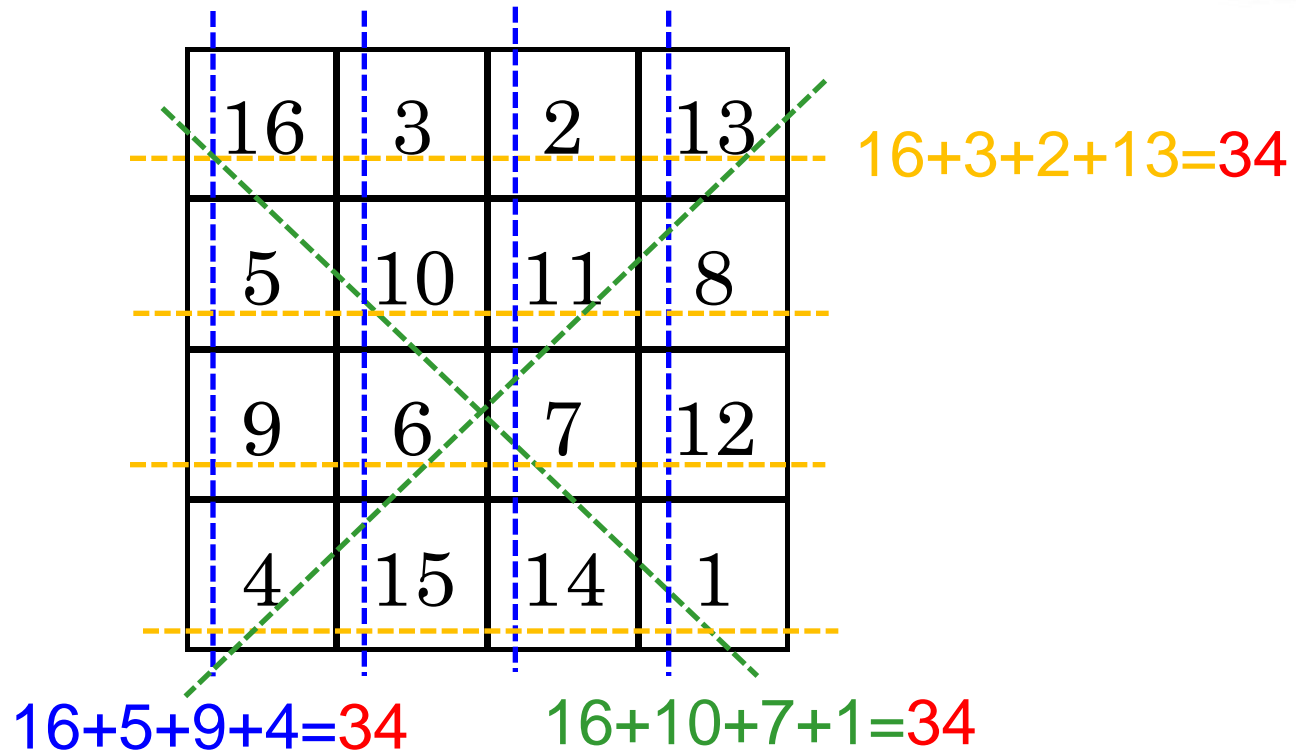
繫神傳曰河出圖洛出書聖人則之又曰天一地二天三地四天五地六天七地八天九地十天數五地數五五位相得而各有合天數二十有五地數三十九天地之數五十有五此所以成變化而行鬼神也此河圖之數也洛書蓋取龜象故其數戴九履一左三右七二四為肩六八為足。  
 蔡元定曰圖書之象自漢孔安國劉歆魏關朗子明有宋康節先生邵雍姜夫皆謂如此至劉牧始兩易其名而諸家因之故今復之悉從其舊。



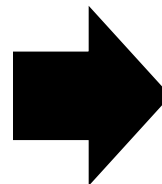
3x3 魔方陣

4	9	2
3	5	7
8	1	6

# Features of Magic Square



- Constant Sum
  - Row, column, diagonal



魔方陣 is Good balance

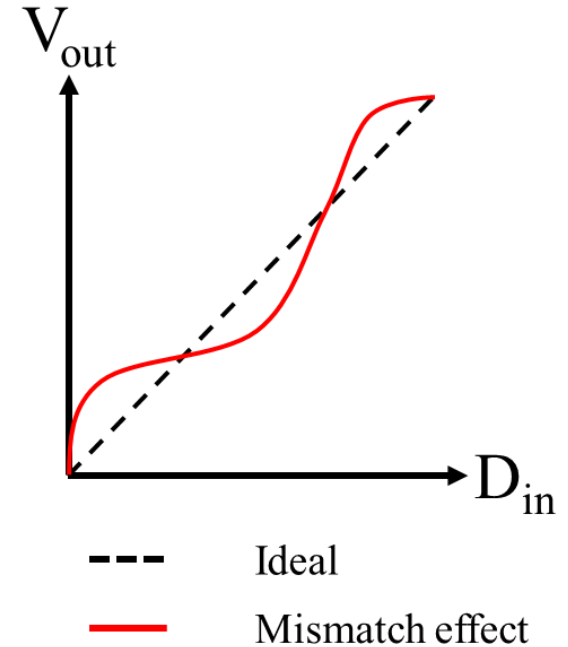
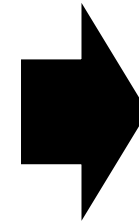
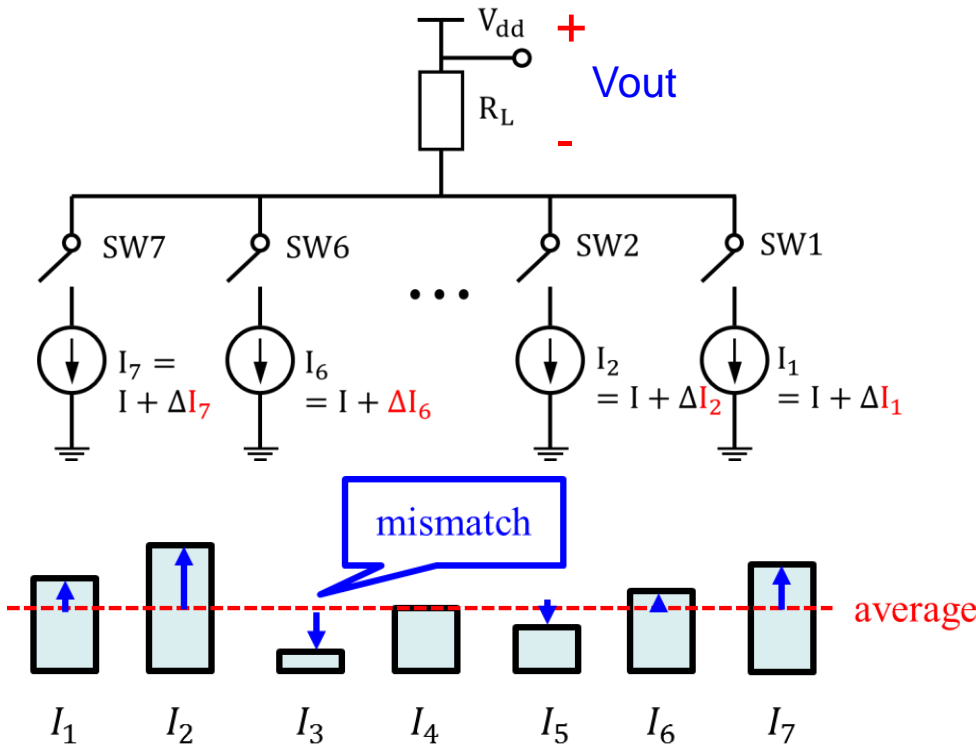


# OUTLINE

- Research Objective
- Current Steering DAC
- What is Magic Square ?
- **Proposed Algorithm**
- Simulation Results
- Conclusion



# Unit Current Source Mismatch Problem

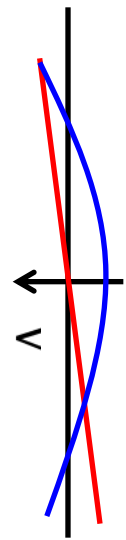
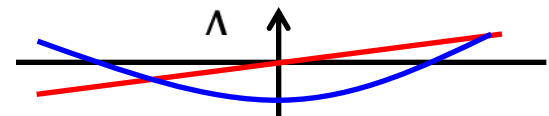
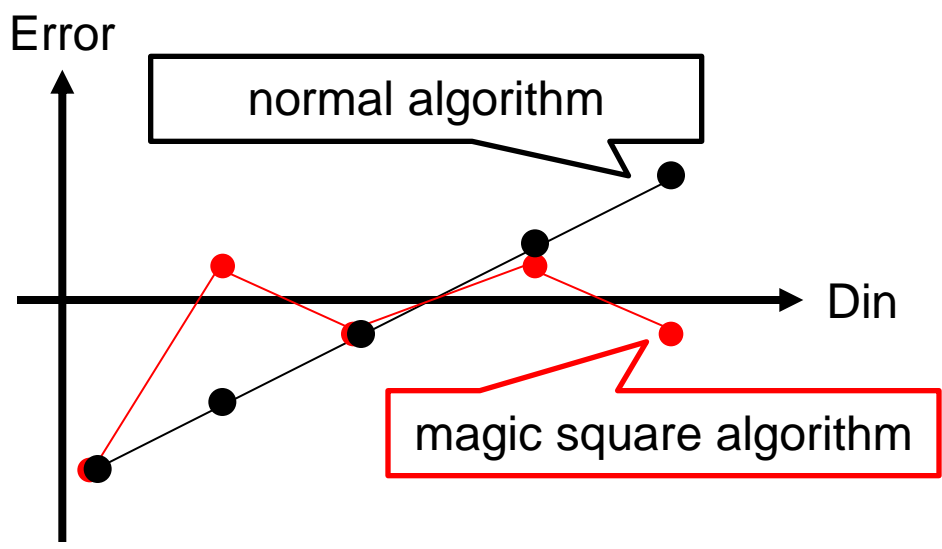


In practice, current sources have mismatches.

➔ **DAC becomes non-linear.**

# Possibility of Using Magic Square

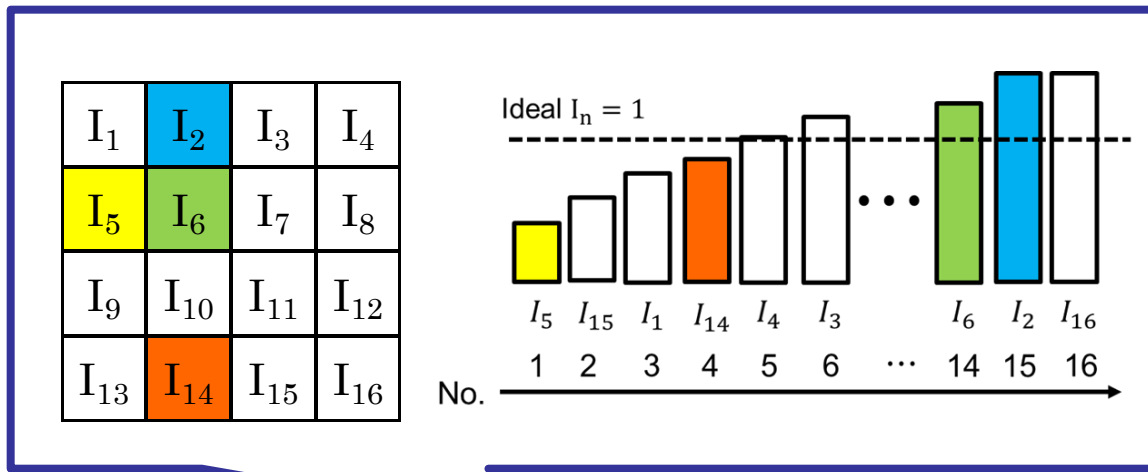
- Semiconductor devices have random and systematic mismatches
- Changing the switching order  
➔ Cancellation of mismatch effects
- We propose magic square algorithm



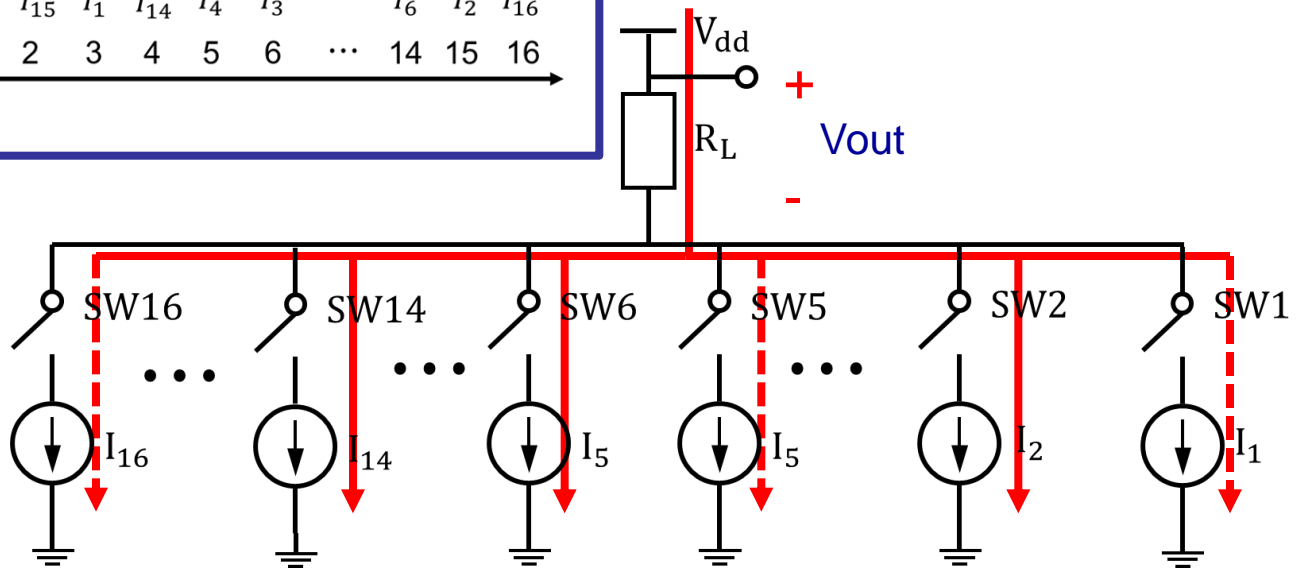
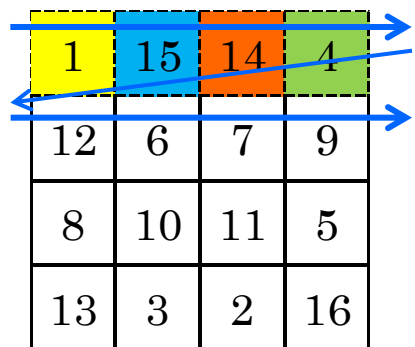
1	15	14	4
12	6	7	9
8	10	11	5
13	3	2	16

# Inspired New Algorithm

- Unit current source selection-order change algorithm
  - Mismatch effect cancellation

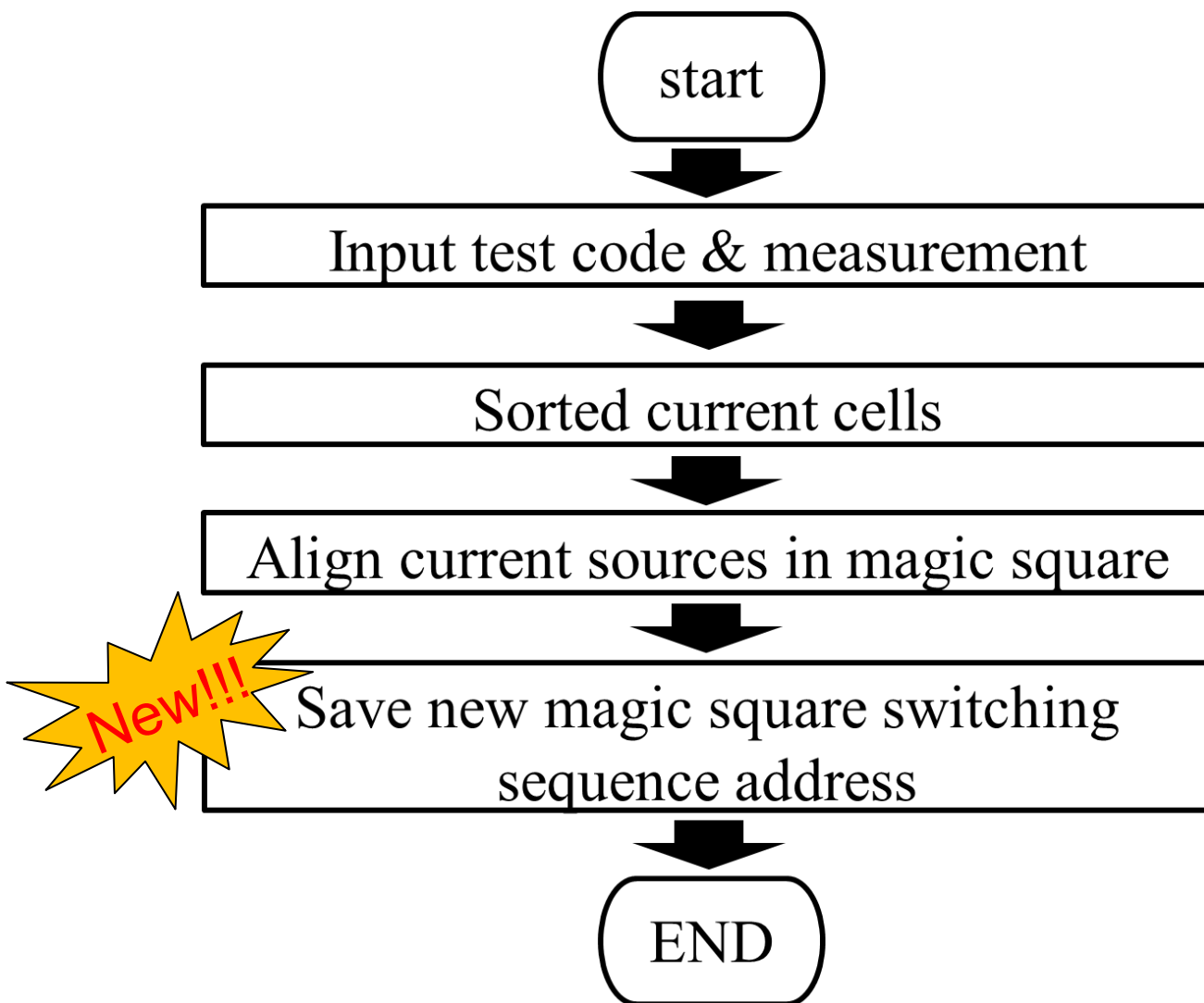


1. Measure the order of unit current cells
2. Align them virtually in magic square
3. Select current cells



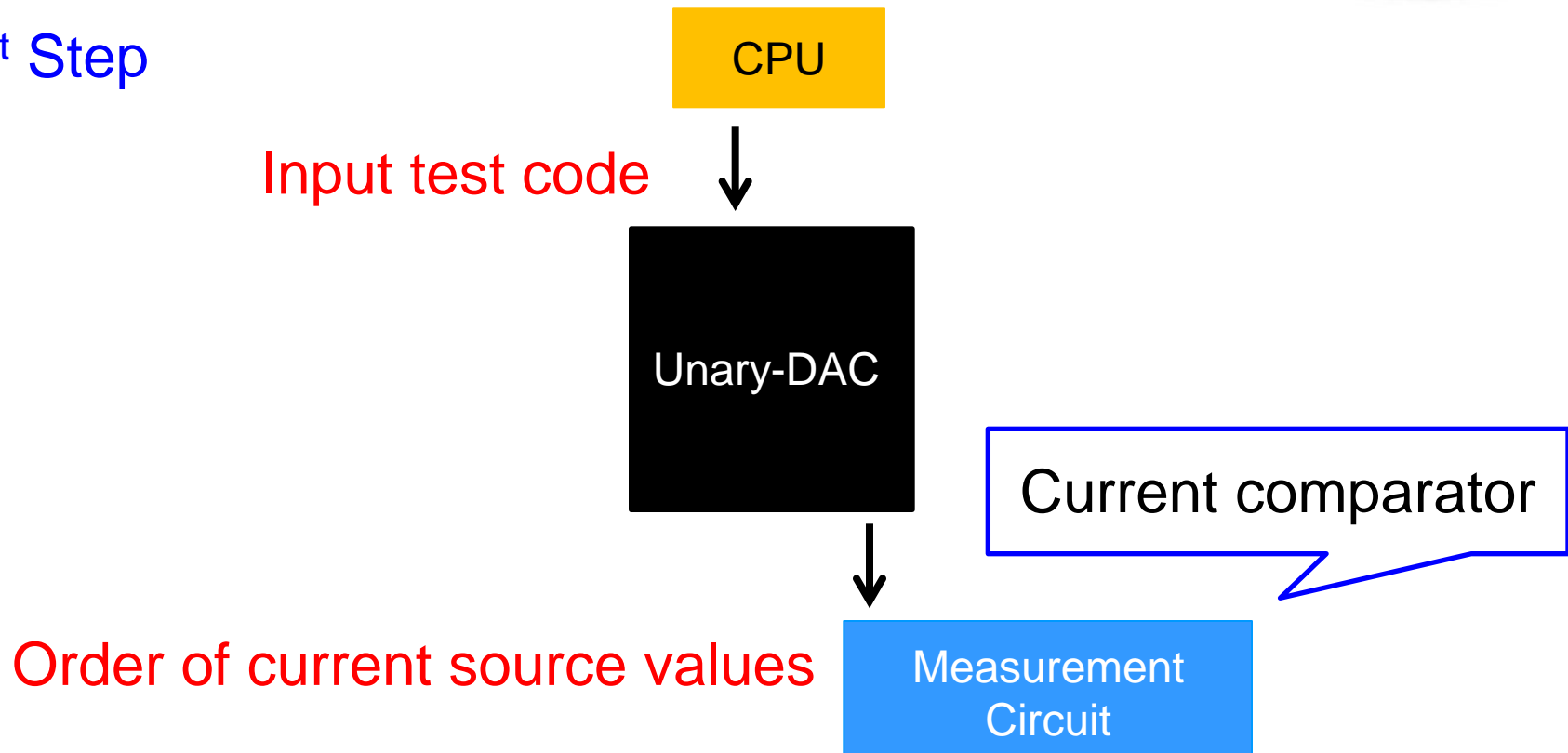
# Proposed DAC Non-linearity

## Calibration Algorithm



## Input Test Code & Measurement

1<sup>st</sup> Step



- CPU => input test code to unary-DAC cells
- Measurement circuit => order of current source values

## Measure Order of Current Cells

### 1<sup>st</sup> Step

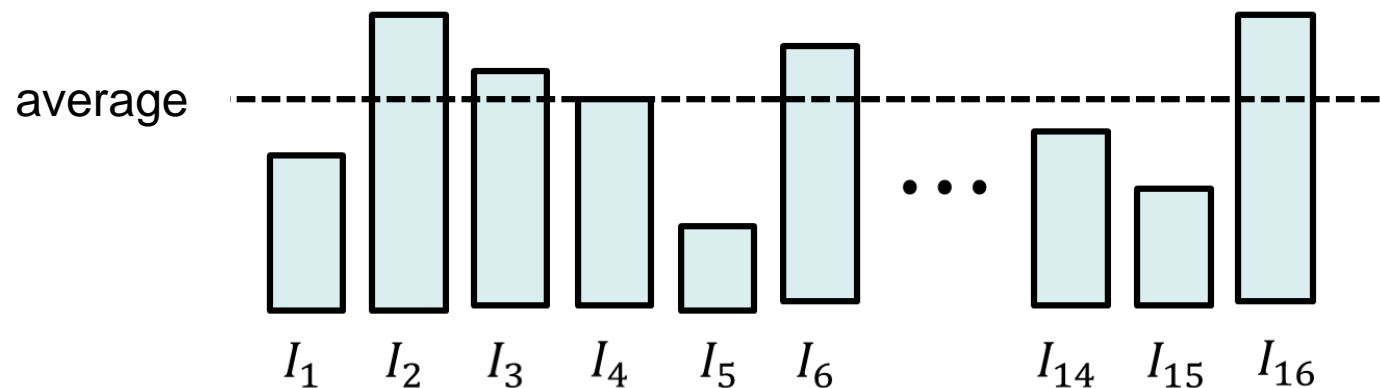
- Measure the order of current cell values by a current comparator.
- Not need accurate value measurement.

### 4-bit case

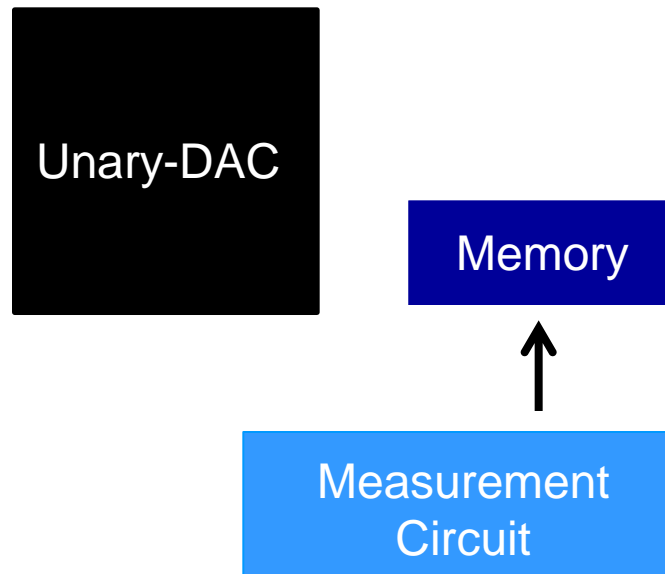
Current cell

$I_1$	$I_2$	$I_3$	$I_4$
$I_5$	$I_6$	$I_7$	$I_8$
$I_9$	$I_{10}$	$I_{11}$	$I_{12}$
$I_{13}$	$I_{14}$	$I_{15}$	$I_{16}$

Original Current Source



## Unit Current Source Sorting

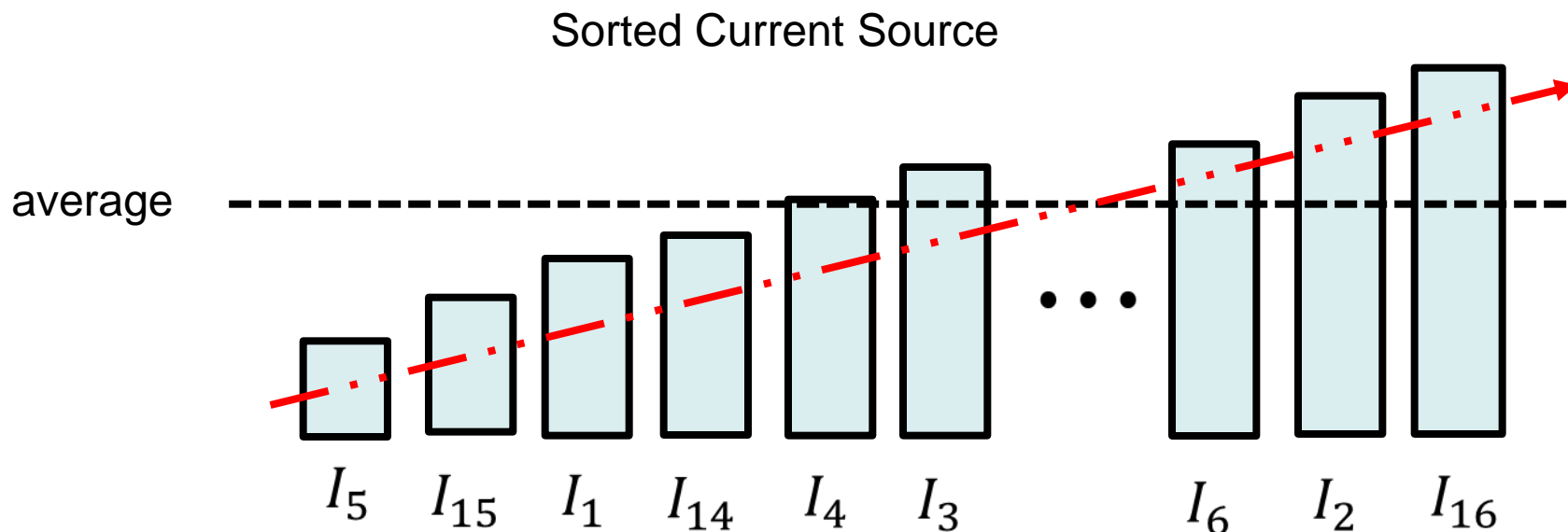
2<sup>nd</sup> step

Sort and store the measured order of the unit current cell values into memory.

## Unit Current Source Sorting

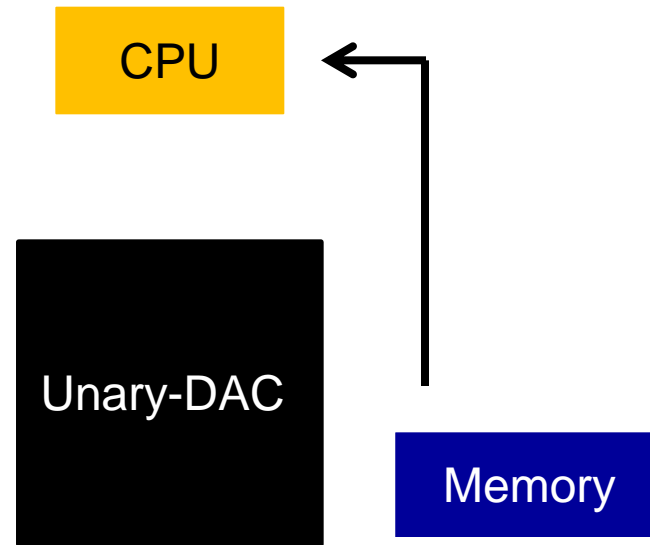
### 2<sup>nd</sup> step

- Sort current source cells ascendingly.
- Store their information of cells number and value into memory.





## Current Source Sorting Based on Magic Square

3<sup>rd</sup> step

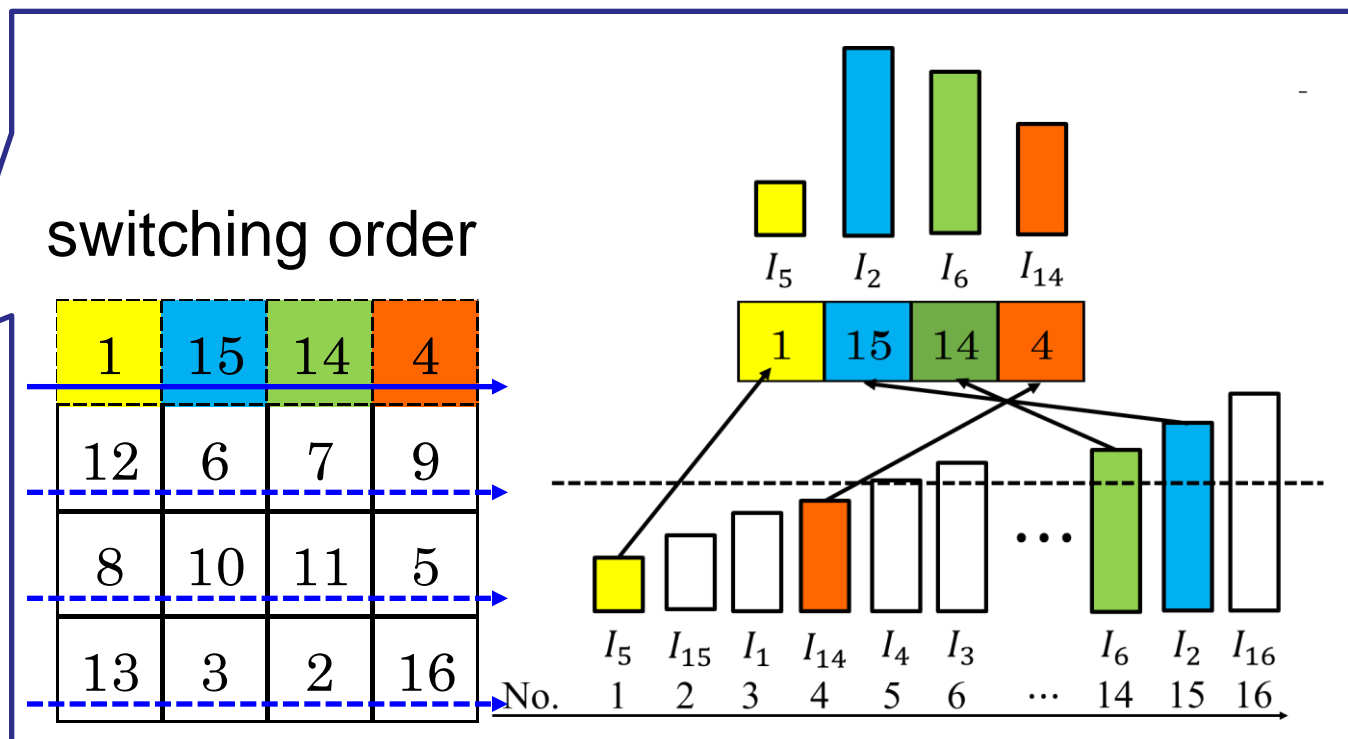
- Re-sort of current source values based on magic square

## Current Source Sorting Based on Magic Square (1)

### 3<sup>rd</sup> step

- Re-sorted of current source values based on magic square
- Store its info in decoder look-up table

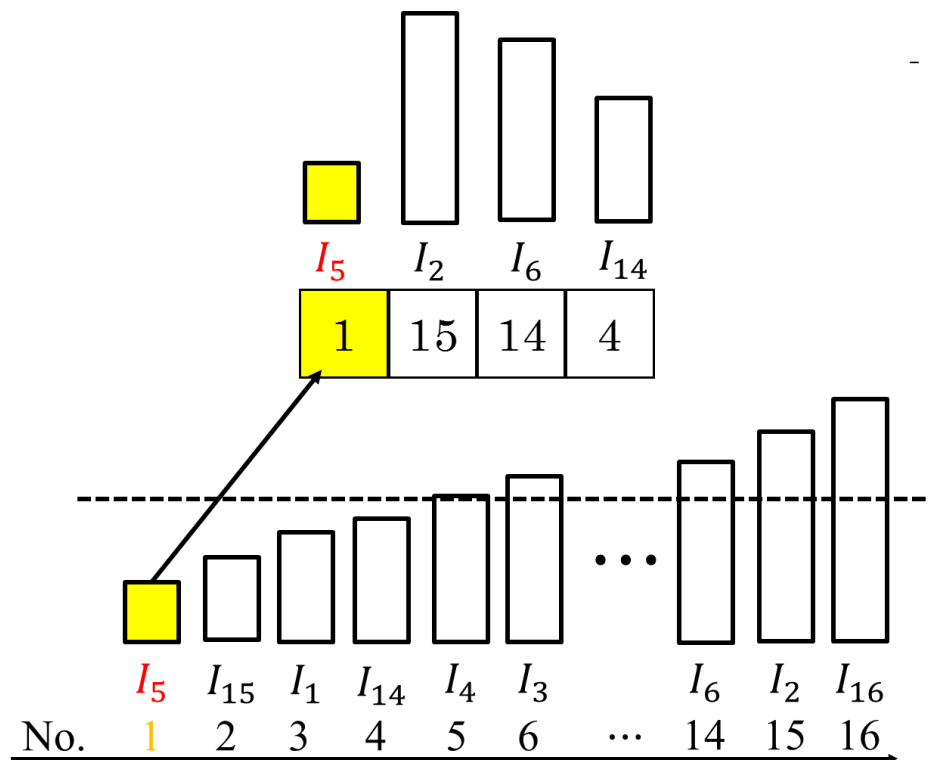
$I_1$	$I_2$	$I_3$	$I_4$
$I_5$	$I_6$	$I_7$	$I_8$
$I_9$	$I_{10}$	$I_{11}$	$I_{12}$
$I_{13}$	$I_{14}$	$I_{15}$	$I_{16}$



## Current Source Sorting Based on Magic Square (2)

- Digital binary input (0001)  
➔ 1 current cells turn on

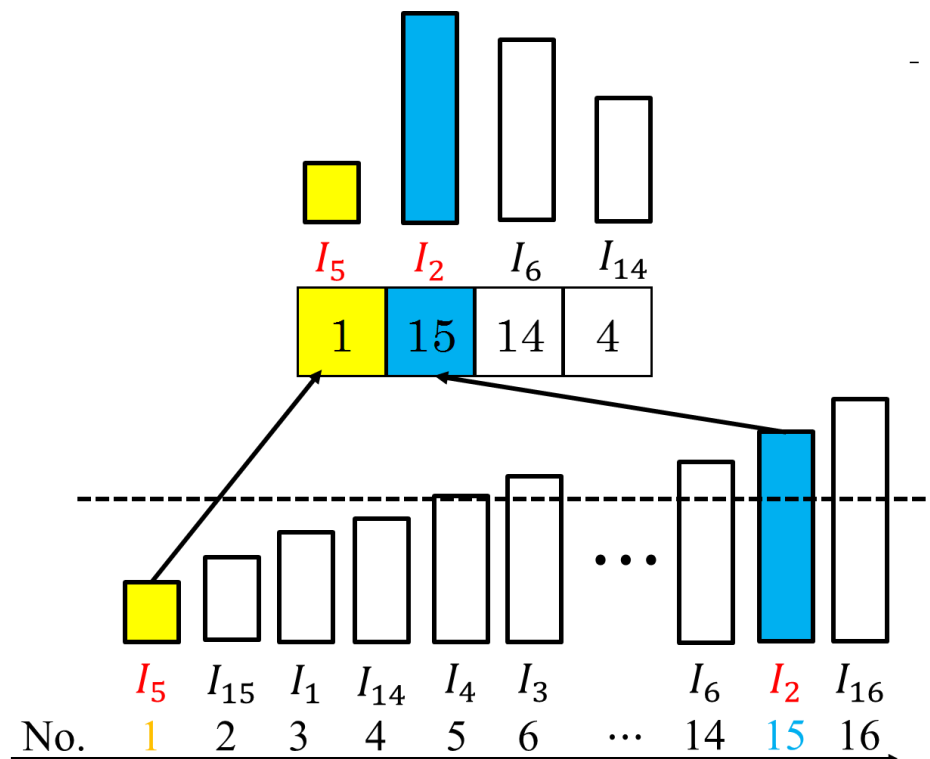
$I_1$	$I_2$	$I_3$	$I_4$
$I_5$	$I_6$	$I_7$	$I_8$
$I_9$	$I_{10}$	$I_{11}$	$I_{12}$
$I_{13}$	$I_{14}$	$I_{15}$	$I_{16}$



## Current Source Sorting Based on Magic Square (3)

- Digital binary input (00**10**)  
➔ 2 current cells turn on

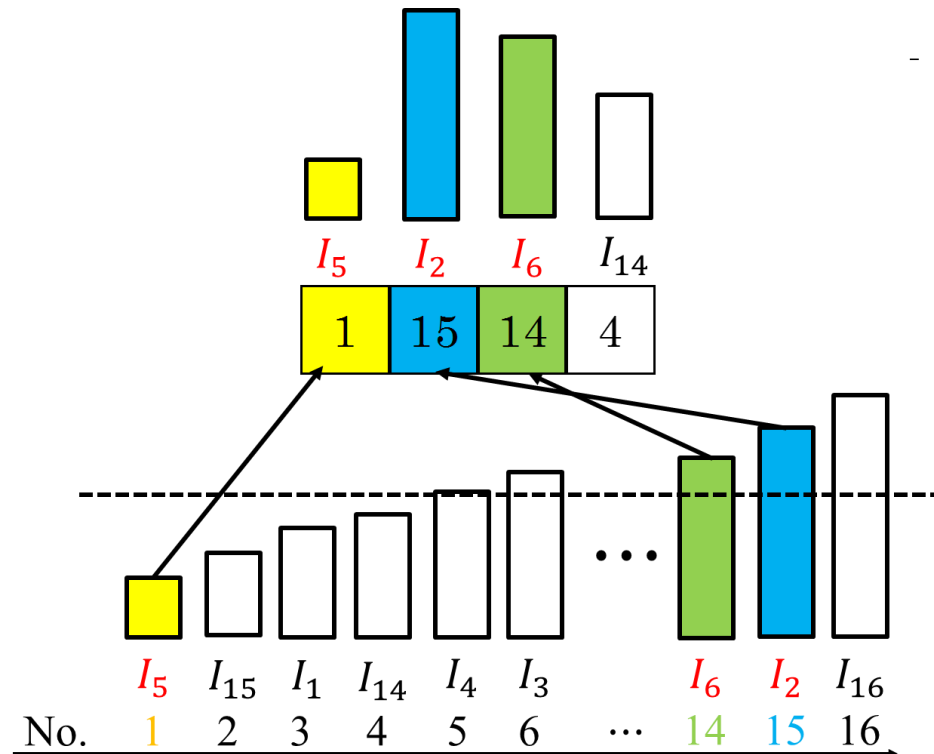
$I_1$	$I_2$	$I_3$	$I_4$
$I_5$	$I_6$	$I_7$	$I_8$
$I_9$	$I_{10}$	$I_{11}$	$I_{12}$
$I_{13}$	$I_{14}$	$I_{15}$	$I_{16}$



## Current Source Sorting Based on Magic Square (4)

- Digital binary input (0011)  
➔ 3 current cells turn on

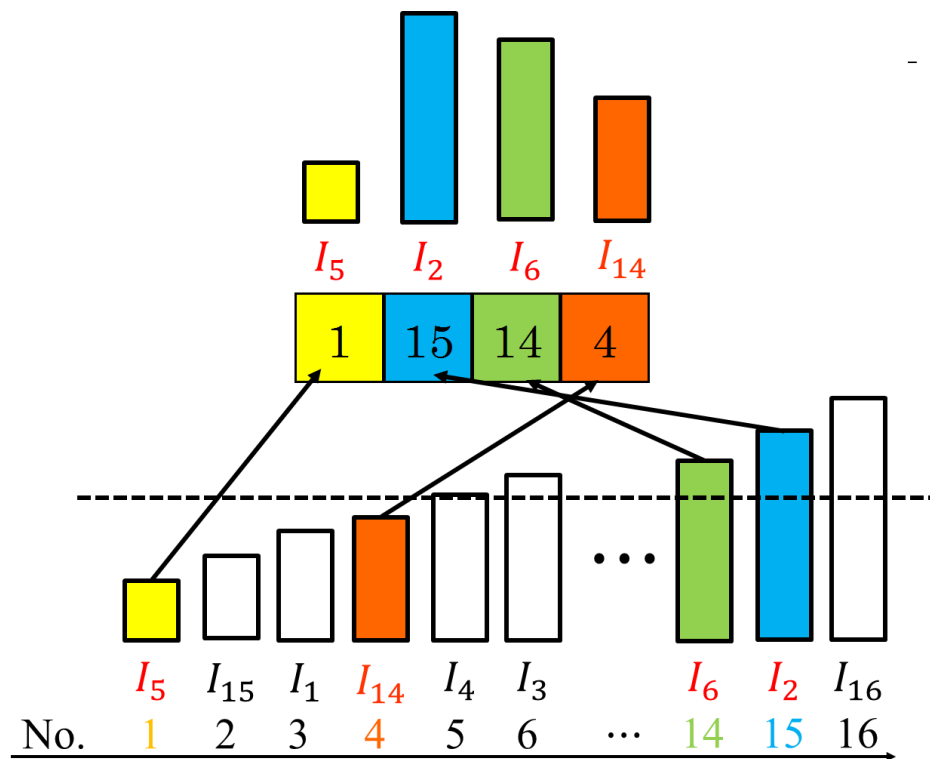
$I_1$	$I_2$	$I_3$	$I_4$
$I_5$	$I_6$	$I_7$	$I_8$
$I_9$	$I_{10}$	$I_{11}$	$I_{12}$
$I_{13}$	$I_{14}$	$I_{15}$	$I_{16}$



## Current Source Sorting Based on Magic Square (5)

- Digital binary input (0100)  
➔ 4 current cells turn on

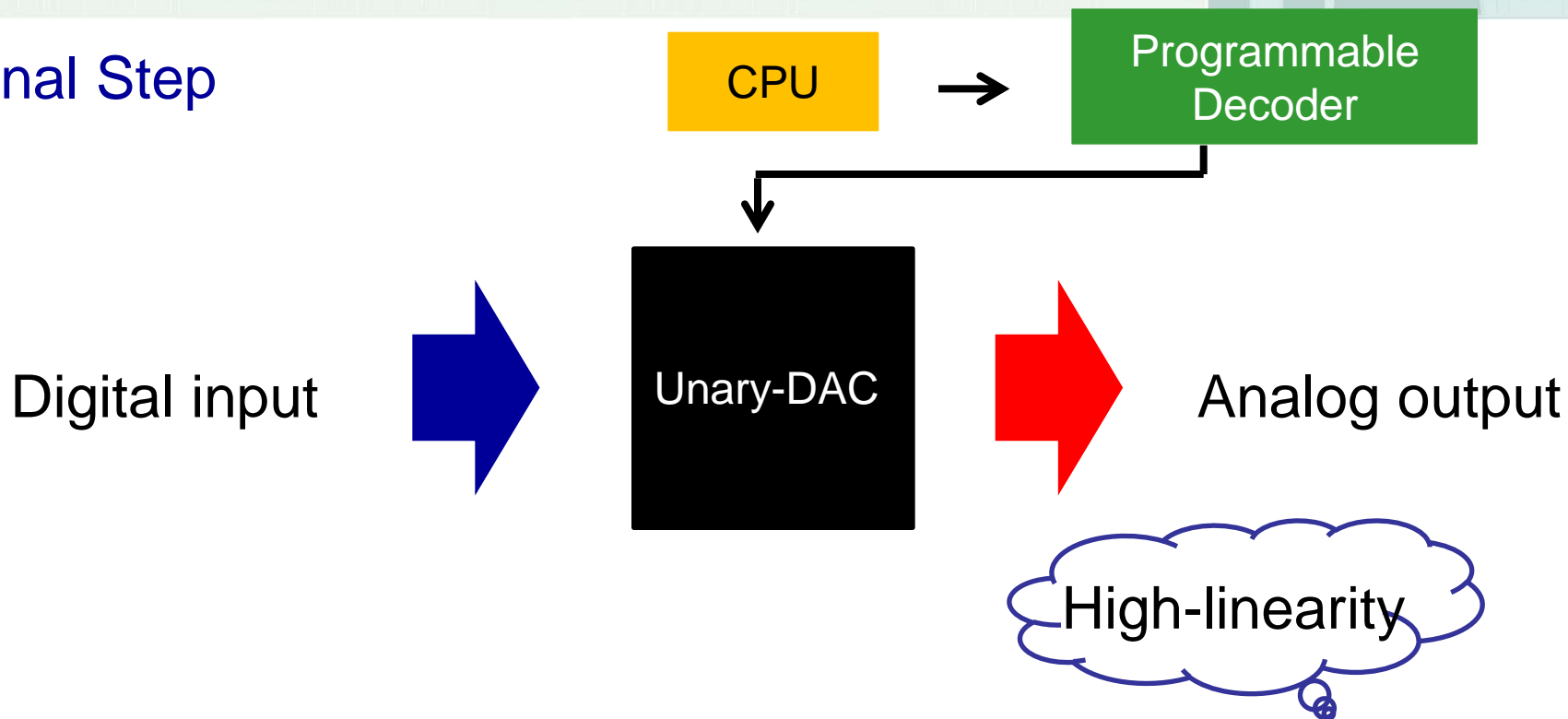
$I_1$	$I_2$	$I_3$	$I_4$
$I_5$	$I_6$	$I_7$	$I_8$
$I_9$	$I_{10}$	$I_{11}$	$I_{12}$
$I_{13}$	$I_{14}$	$I_{15}$	$I_{16}$



# Proposed Algorithm

## LUT-Magic Square Decoder

Final Step



Store switching sequence based on magic square into programmable decoder.





# OUTLINE

- Research Objective
- Current Steering DAC
- What is Magic Square ?
- Proposed Algorithm
- **Simulation Results**
- Conclusion

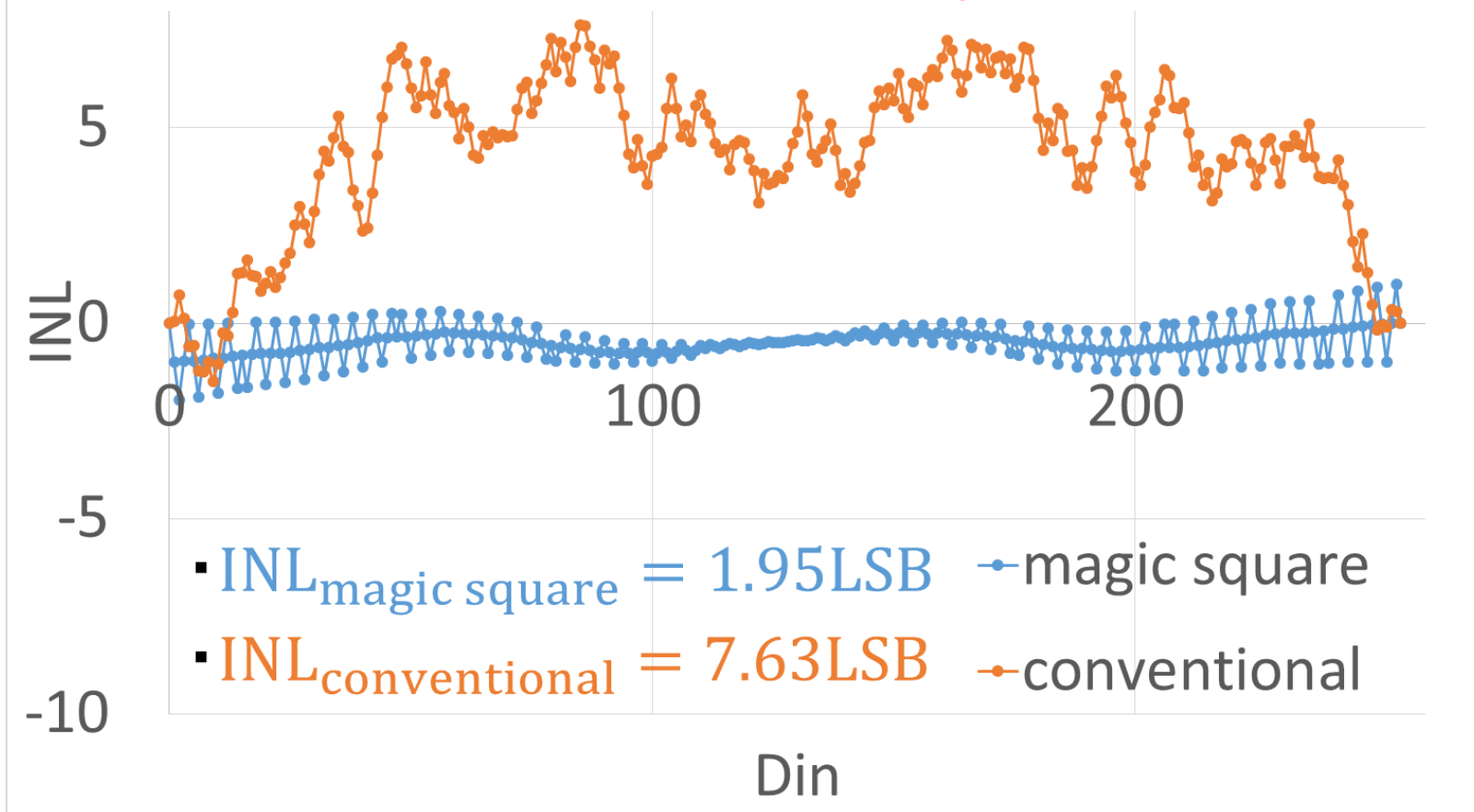
# Simulation Conditions

- MATLAB simulation
- 8-bit unary DAC
  - Static performance (INL, DNL)
  - Dynamic performance (SFDR)
- Compared two methods
  - Conventional thermometer-code decoder usage
  - Proposed magic-square-based algorithm
- Mismatch of current sources
  - Current sources have average of value 1.0
  - Random number between  $-1 < \text{mismatch} < +1$  (uniform distribution)

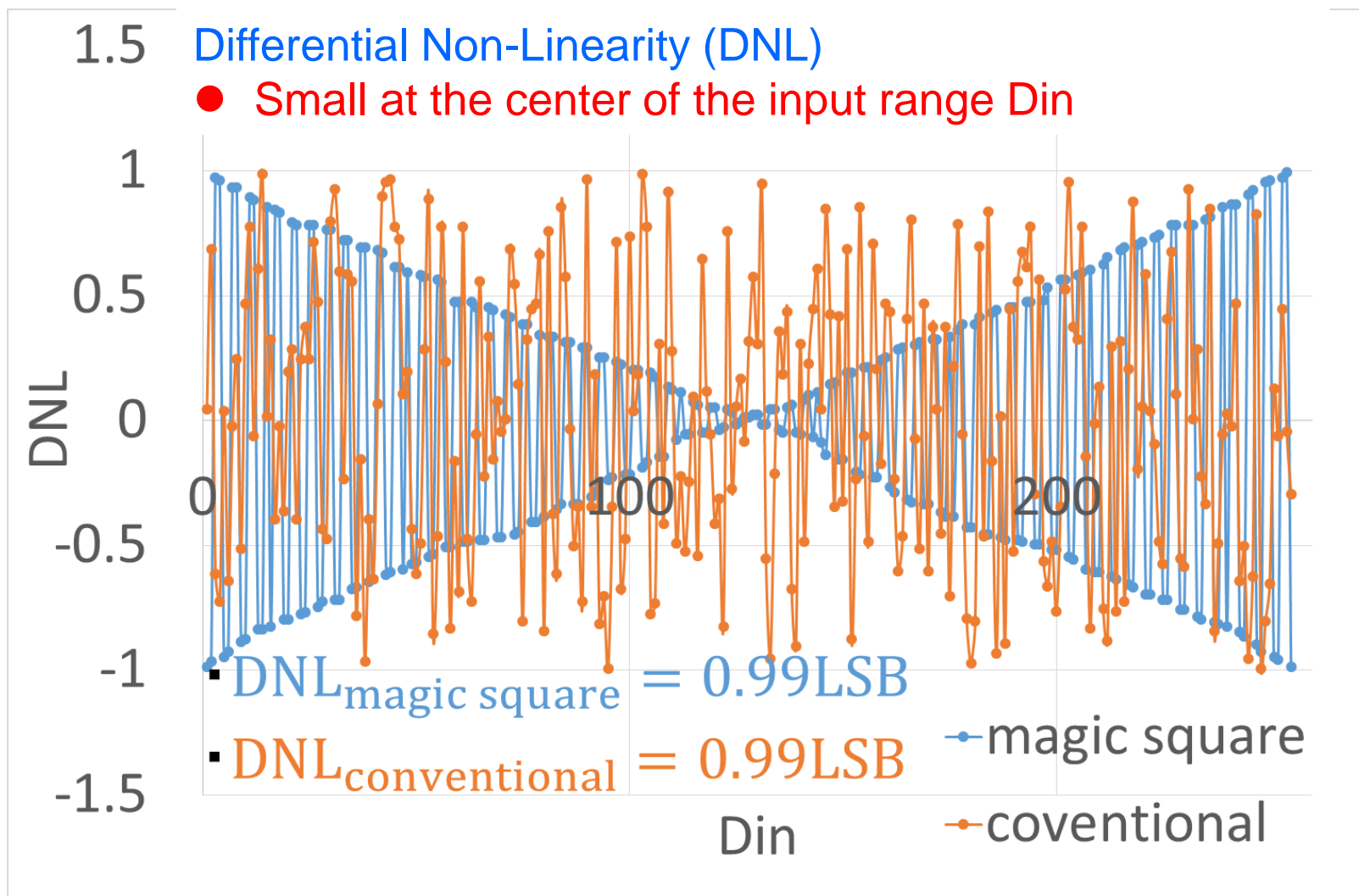
## - Static Performance INL -

### Integral Non-Linearity (INL)

- 5.7 LSB improvement by the magic square algorithm
- 0.0 LSB at the center of the input range  $D_{in}$

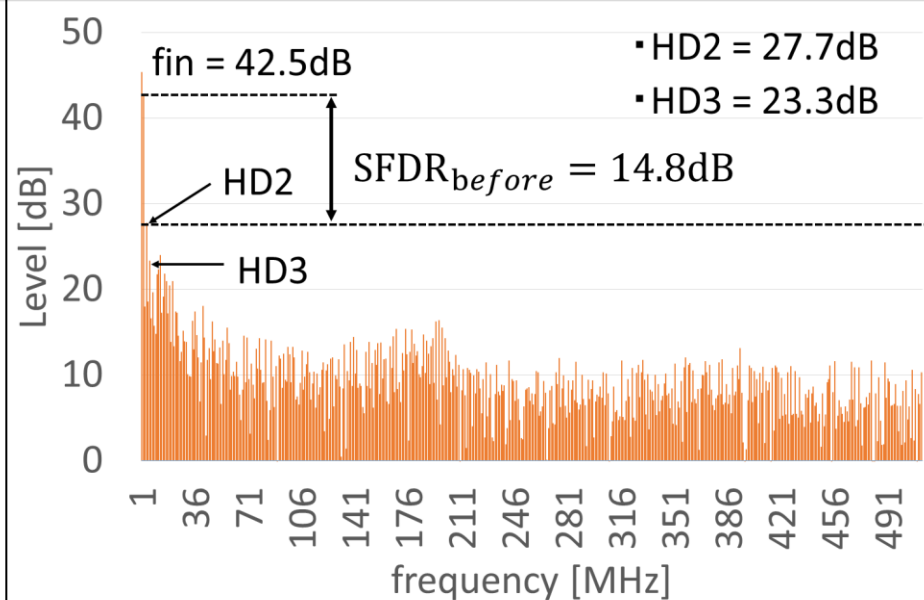


## - Static Performance DNL -

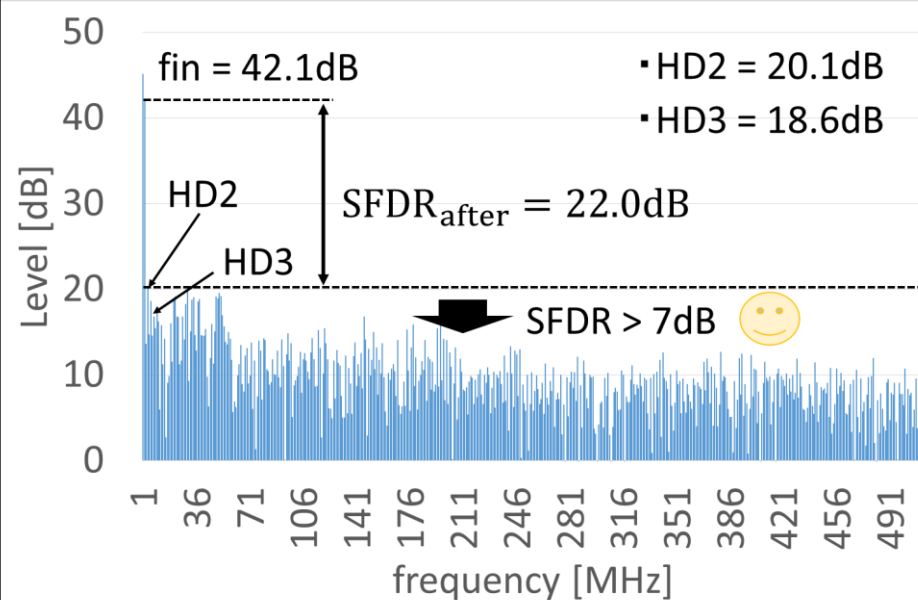


## - Dynamic Performance SFDR -

### ● Conventional method



### ● Proposed method



● SFDR improvement by 7 dB

# OUTLINE

- Research Objective
- Current Steering DAC
- What is Magic Square ?
- Proposed Algorithm
- Simulation Results
- **Conclusion**

## Conclusion

- Unary DAC linearity improvement
  - Cancel unit current cell mismatch effects
  - Unit current cell selection algorithm
    - ➔ Digital method
  - Based on magic square
  - Measurement of the order of current cell values
- MATLAB simulation
  - INL , DNL improvement
    - at the center of the input range.
  - SFDR improvement



# Final Statement

## 温故知新

Classical mathematics can contribute modern technology.

