IEEE International Symposium on Intelligent Signal Processing and Communication Systems 2017

NOVEMBER 6-9, 2017, XIAMEN, CHINA

Nov. 8 WP-L6 16:10-17:40

DAC Linearity Improvement With Layout Technique Using Magic and Latin Squares

Dan Yao, Yifei Sun, M. Higashino, S. N. Mohyar T. Yanagida

T. Arafune, N. Tsukiji, H. Kobayashi

Gunma University

Contents

- Research Objective
- Segment Type DA Converter
- Characteristic of Variation in Circuit Element
- Proposed Layout Method
  - Magic Square
  - Latin Square
- Conclusion

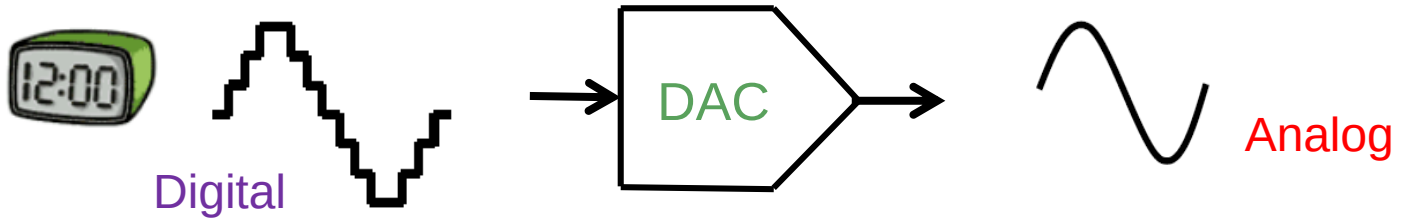
Contents

Research Objective

Objective

- Development of

  a highly linear digital-to-analog converter (DAC)



Our Approach

- DAC layout technique

  to cancel systematic mismatch effects

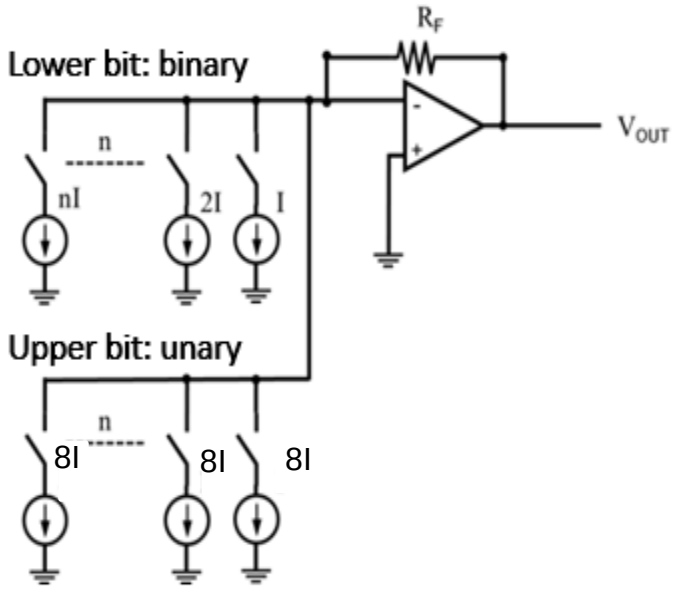
  among unit current cells.

  -Layout based on Magic and Latin Squares

New!!

Contents

Segment Type DAC Configuration



✓ Binary (Lower bits)

- Small circuit
- Large glitch
- Large mismatch effect & Large nonlinearity

✓ Unary (Upper bits)

- Large circuit
- Small glitch
- Small mismatch effect & modest linearity

Segmented DAC

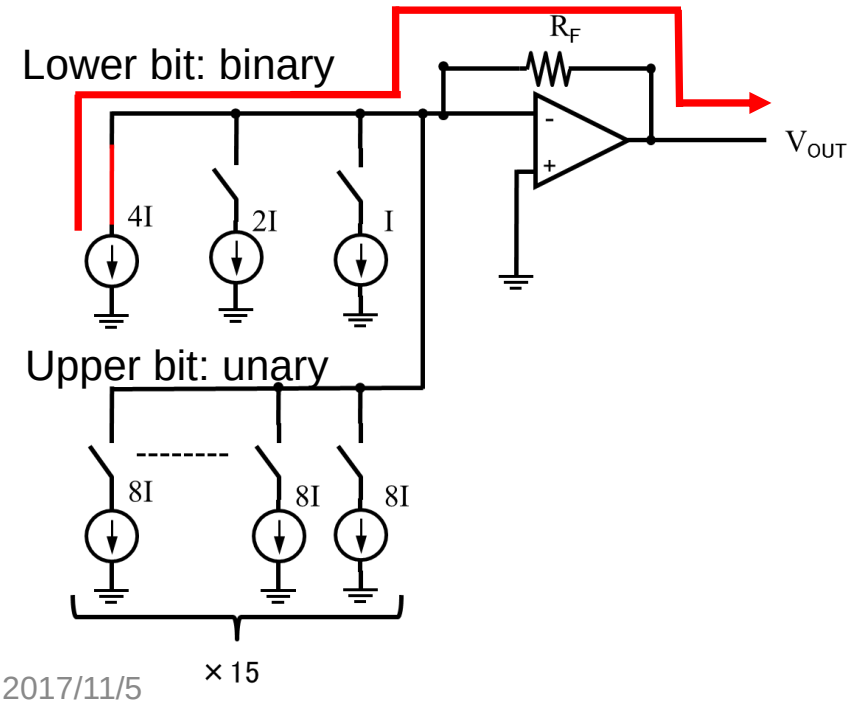
Focus !!

Segment Type DAC (7-bit case)

ex.1

In case digital input =4

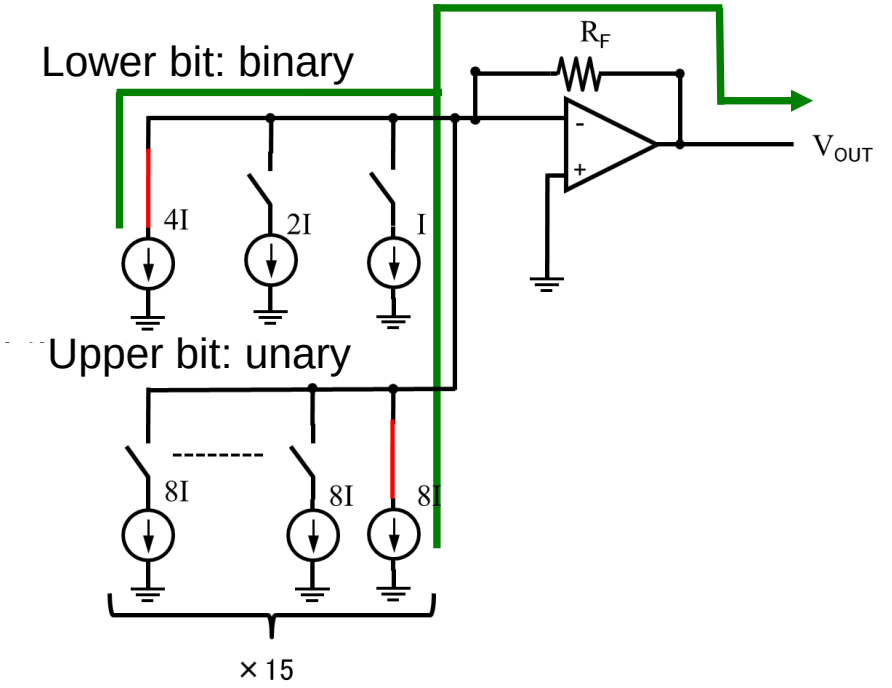
(0000100) → $V_{out} = 4IR_F$



ex.2

In case digital input =12

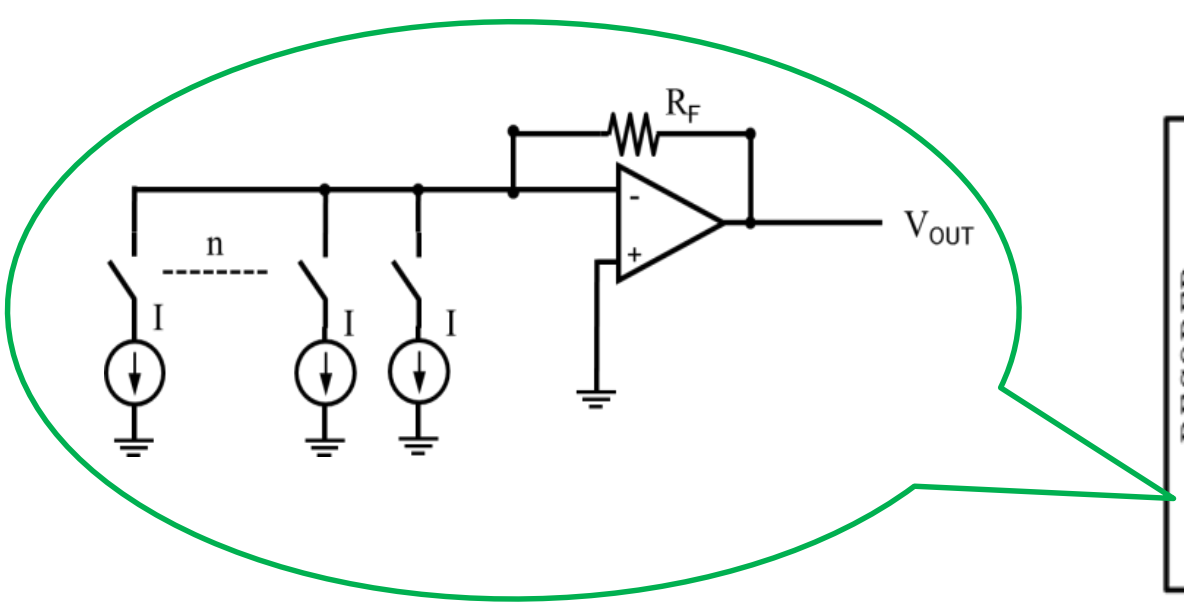
(0001100) → $V_{out} = 12IR_F$



Unary DAC Features

- Identical current sources
- Small glitch
- Inherent monotonicity

- Large circuits
  - Decoder
  - Many switches and current sources



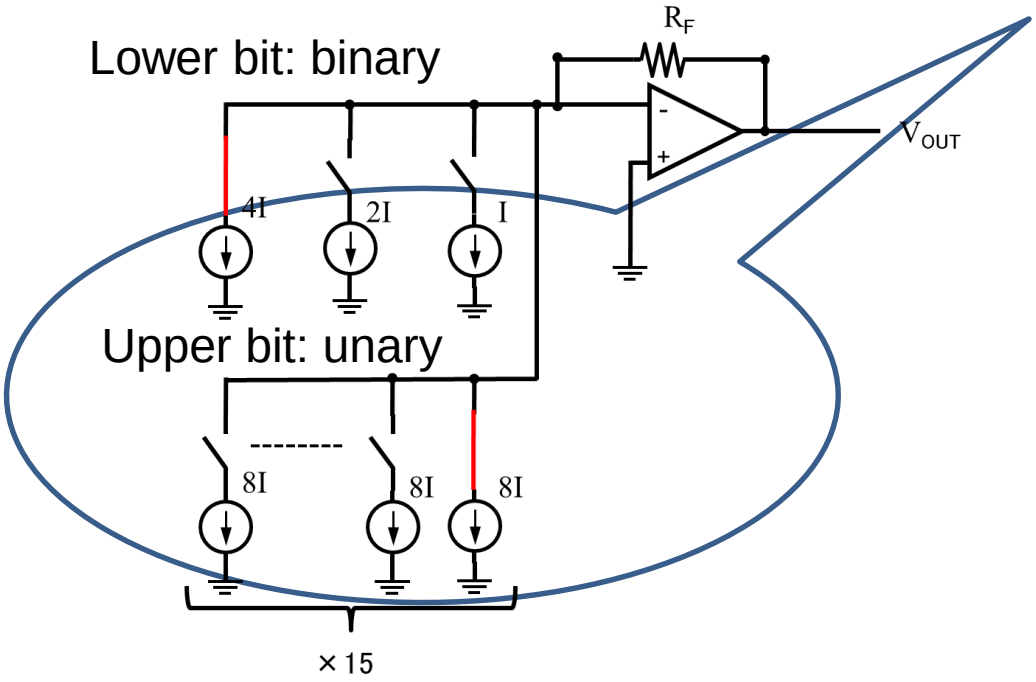
DECODER

DECODER			
S1	S2	S3	S4
S5	S6	S7	S8
S9	S10	S11	S12
S13	S14	S15	S16

Unary DAC Current Cells Layout

✓ 7bit DA Converter

(0001100) → $V_{out} = 12IR_F$

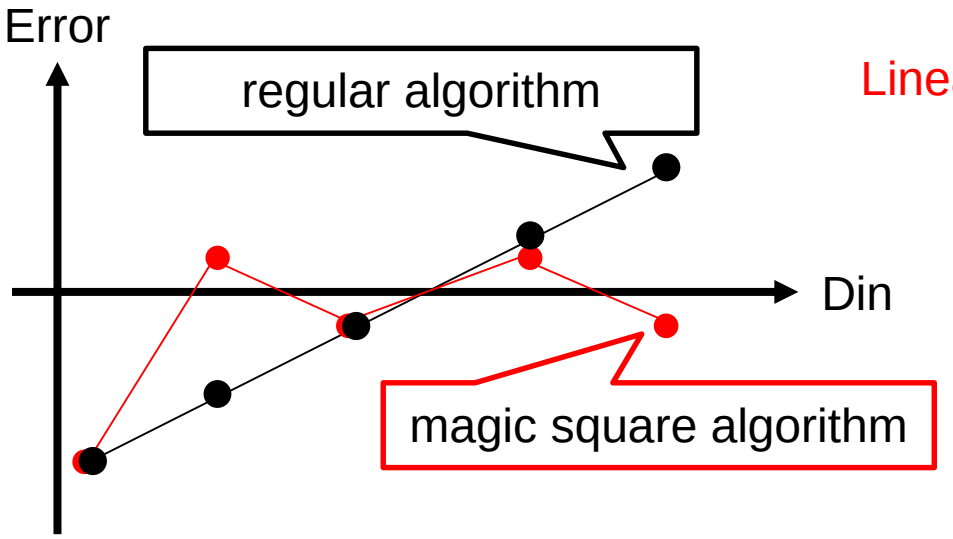


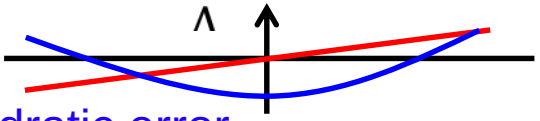
S1	S2	S3	S4
S5	S6	S7	S8
S9	S10	S11	S12
S13	S14	S15	S16

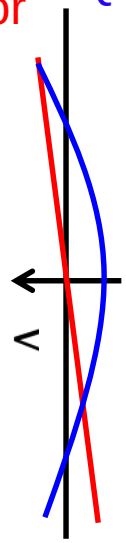
Unit current cell (unary)

Cell Layout and Systematic Mismatch

- Semiconductor devices have systematic mismatches
- Changing the unit cell layout order
  → Cancellation of systematic mismatch effects
- We propose magic and Latin squares algorithms







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

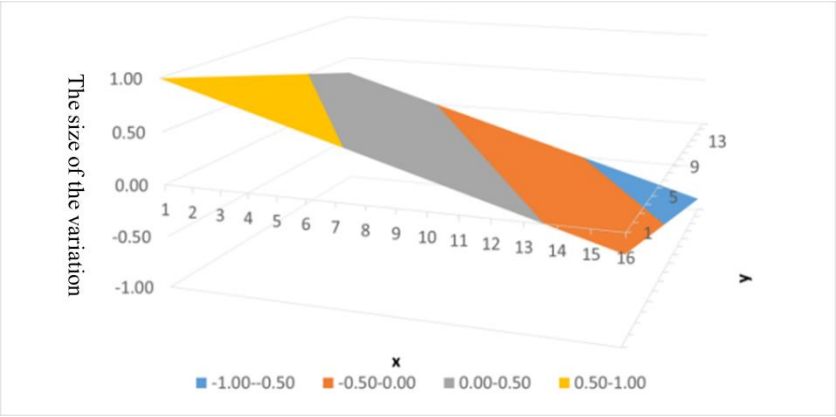
Contents

Variation in Circuit Element Characteristics

◆ Systematic variations

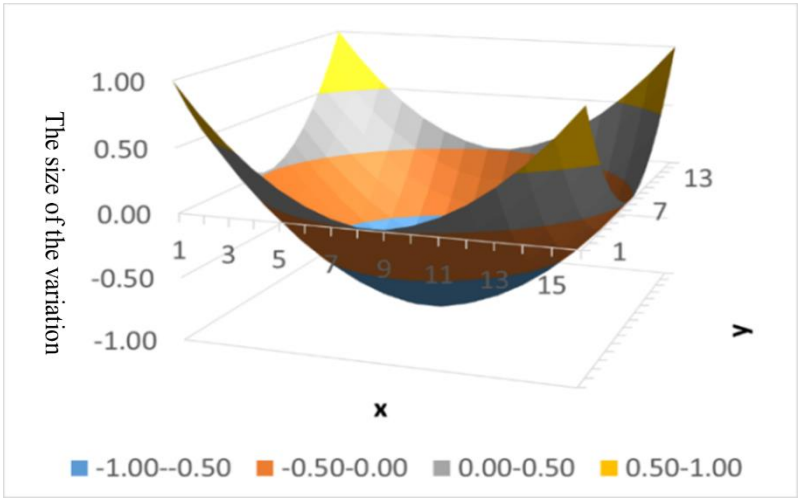
- ✓ Voltage drop
- ✓ Thickness of oxide film
- ✓ Doping

→ Linear error



- ✓ Mechanical stress
- ✓ Temperature distribution
- ✓ In wafer plane

→ Quadratic error



Joint Error (Sum of both)

Systematic Variation Model

Linear Error

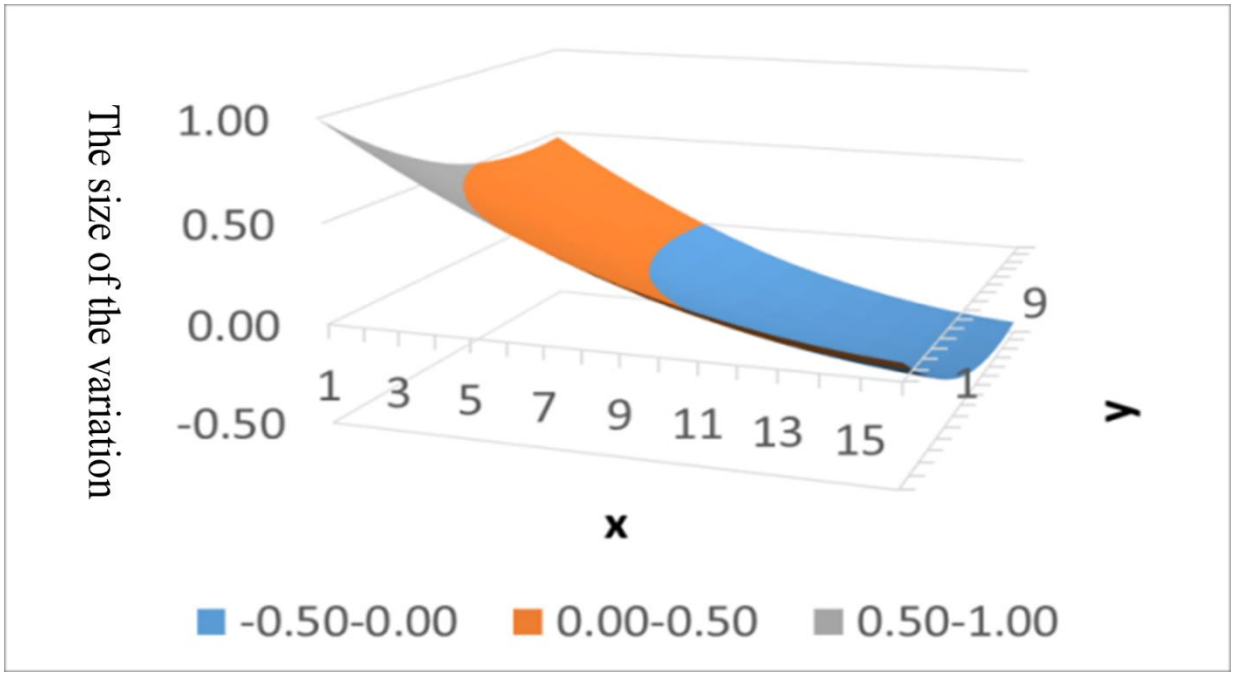
$$\varepsilon_l(x,y) = g_l * \cos\theta * x + g_l * \sin\theta * y$$

Quadratic Error

$$\varepsilon_q(x,y) = g_q * (x^2 + y^2) - a_0$$

Joint Errors

$$\varepsilon_j(x,y) = \varepsilon_l(x,y) + \varepsilon_q(x,y)$$

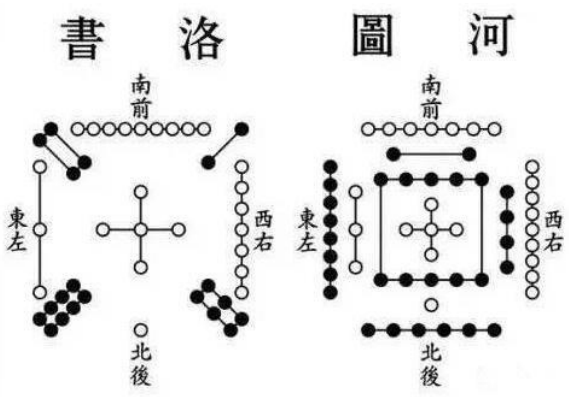


Contents

- Research Objective
- Segment Type DA Converter
- Characteristic of Variation in Circuit Element
- Proposed Method
  - Magic Square
  - Latin Square
- Conclusion

What is Magic Square ?

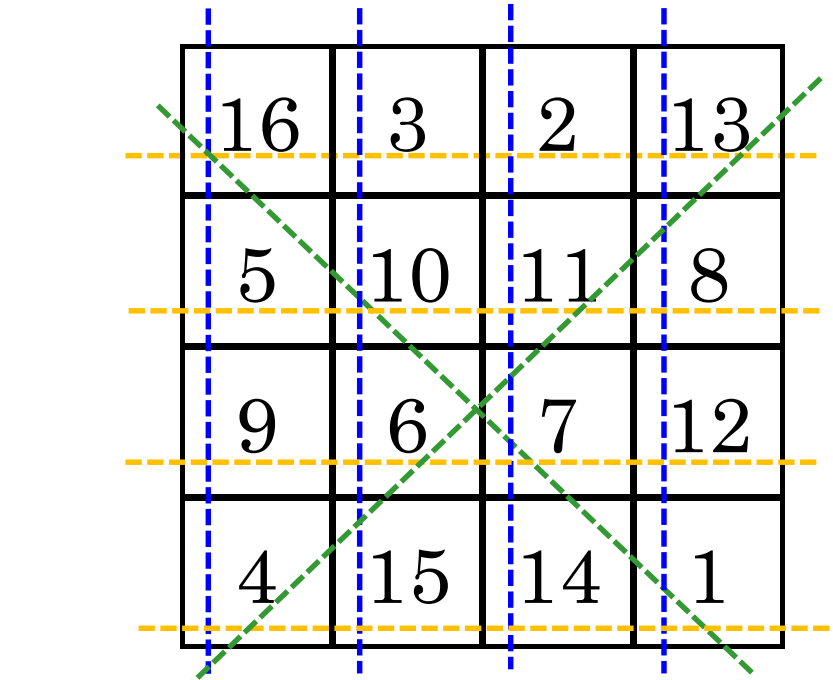
- Classical mathematics
- Origin from Chinese academia
- “Constant sum” characteristics
- Varieties of magic squares



3 × 3 Magic Square

2	9	4
7	5	3
6	1	8

Magic Square Features



Constant Sum
Row, Column, Diagonal

→

Magic Square is good balance

Magic Square for Layout Algorithm

✓ Concentric Magic Square

Even if one side is removed from the outside,
it does not lose compatibility

Numbers are
in symmetrical positions

expected

Effective for cancellation of
systematic variation effects

59	5	4	62	63	1	8	58
9	18	17	49	50	42	19	56
55	20	28	33	29	40	45	10
54	44	38	31	35	26	21	11
12	43	39	30	34	27	22	53
13	24	25	36	32	37	41	52
51	46	48	16	15	23	47	14
7	60	61	3	2	64	57	6

Magic Square for 16x16 Cell Layout

- Concentric Magic Square

8-bit unit current source cells
by combining 8-th order squares

59	5	4	62	63	1	8	58
9	18	17	49	50	42	19	56
55	20	28	33	29	40	45	10
54	44	38	31	35	26	21	11
12	43	39	30	34	27	22	53
13	24	25	36	32	37	41	52
51	46	48	16	15	23	47	14
7	60	61	3	2	64	57	6

A1	B1
B2	A2

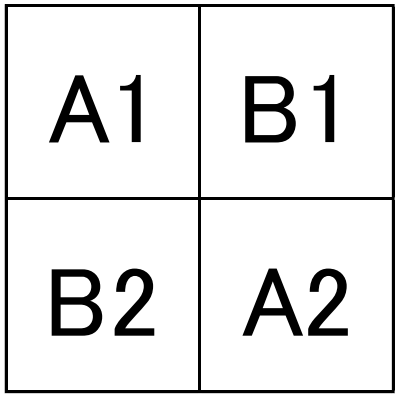
A: Magic square

B: 45 ° counterclockwise rotation

16x16 Cell Layout Details

◆ Concentric Magic Square

- algorithm

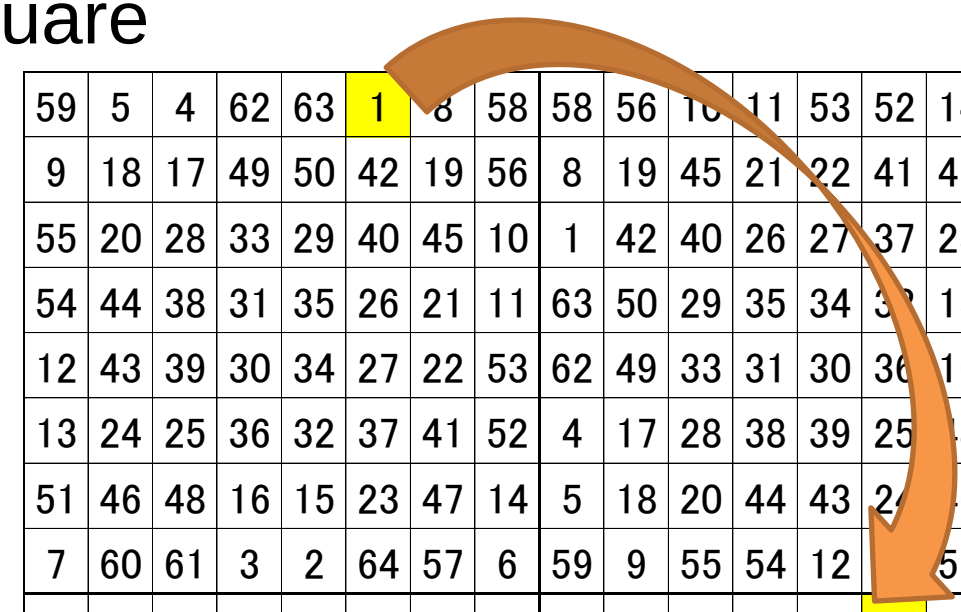


59	5	4	62	63	1	8	58	58	56	10	11	53	52	14	6
9	18	17	49	50	42	19	56	8	19	45	21	22	41	47	57
55	20	28	33	29	40	45	10	1	42	40	26	27	37	23	64
54	44	38	31	35	26	21	11	63	50	29	35	34	32	15	2
12	43	39	30	34	27	22	53	62	49	33	31	30	36	16	3
13	24	25	36	32	37	41	52	4	17	28	38	39	25	48	61
51	46	48	16	15	23	47	14	5	18	20	44	43	24	46	60
7	60	61	3	2	64	57	6	59	9	55	54	12	13	51	7
58	56	10	11	53	52	14	6	59	5	4	62	63	1	8	58
8	19	45	21	22	41	47	57	9	18	17	49	50	42	19	56
1	42	40	26	27	37	23	64	55	20	28	33	29	40	45	10
63	50	29	35	34	32	15	2	54	44	38	31	35	26	21	11
62	49	33	31	30	36	16	3	12	43	39	30	34	27	22	53
4	17	28	38	39	25	48	61	13	24	25	36	32	37	41	52
5	18	20	44	43	24	46	60	51	46	48	16	15	23	47	14
59	9	55	54	12	13	51	7	7	60	61	3	2	64	57	6

16x16 Cell Layout Details

- Concentric Magic Square
- Unit cell selection algorithm

A1	B1
B2	A2

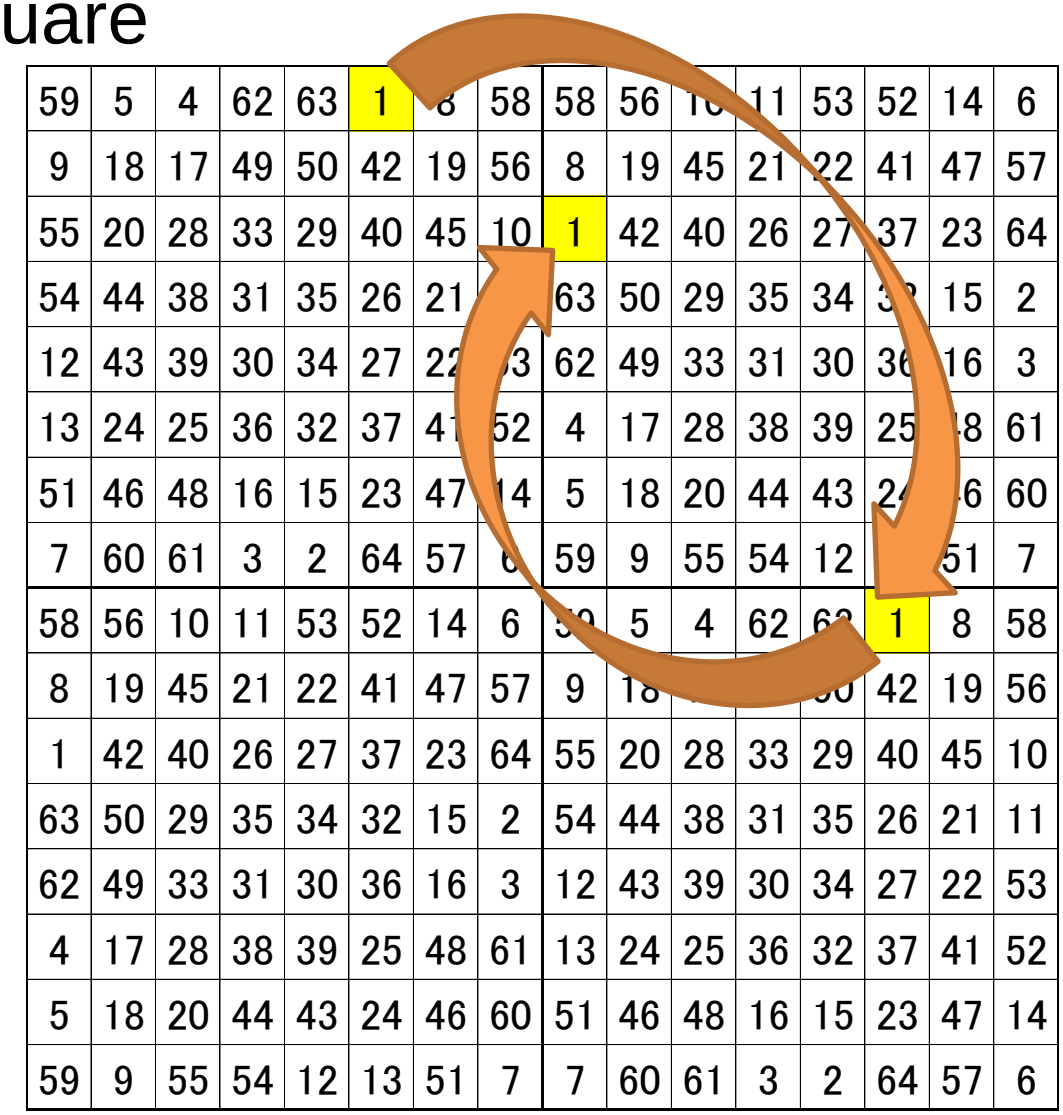


59	5	4	62	63	1	8	58	58	56	10	11	53	52	14	6
9	18	17	49	50	42	19	56	8	19	45	21	22	41	47	57
55	20	28	33	29	40	45	10	1	42	40	26	27	37	23	64
54	44	38	31	35	26	21	11	63	50	29	35	34	3	15	2
12	43	39	30	34	27	22	53	62	49	33	31	30	36	16	3
13	24	25	36	32	37	41	52	4	17	28	38	39	25	48	61
51	46	48	16	15	23	47	14	5	18	20	44	43	24	46	60
7	60	61	3	2	64	57	6	59	9	55	54	12	13	51	7
58	56	10	11	53	52	14	6	59	5	4	62	63	1	8	58
8	19	45	21	22	41	47	57	9	18	17	49	50	42	19	56
1	42	40	26	27	37	23	64	55	20	28	33	29	40	45	10
63	50	29	35	34	32	15	2	54	44	38	31	35	26	21	11
62	49	33	31	30	36	16	3	12	43	39	30	34	27	22	53
4	17	28	38	39	25	48	61	13	24	25	36	32	37	41	52
5	18	20	44	43	24	46	60	51	46	48	16	15	23	47	14
59	9	55	54	12	13	51	7	7	60	61	3	2	64	57	6

16x16 Cell Layout Details

- Concentric Magic Square
- Unit cell selection algorithm

A1	B1
B2	A2

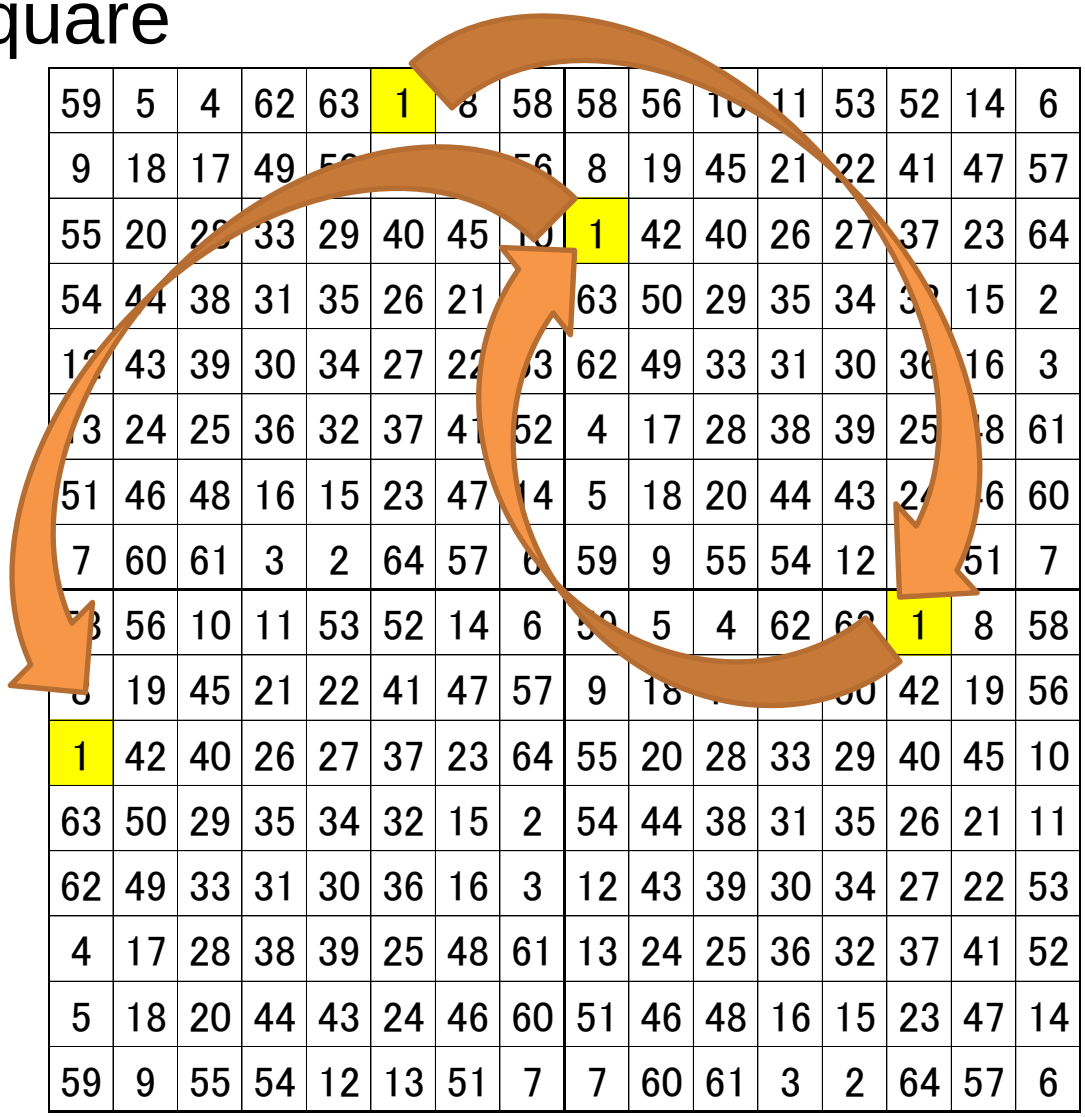


59	5	4	62	63	1	3	58	58	56	10	11	53	52	14	6
9	18	17	49	50	42	19	56	8	19	45	21	22	41	47	57
55	20	28	33	29	40	45	10	1	42	40	26	27	37	23	64
54	44	38	31	35	26	21	11	63	50	29	35	34	32	15	2
12	43	39	30	34	27	22	3	62	49	33	31	30	36	16	3
13	24	25	36	32	37	41	52	4	17	28	38	39	25	48	61
51	46	48	16	15	23	47	14	5	18	20	44	43	24	46	60
7	60	61	3	2	64	57	6	59	9	55	54	12	13	51	7
58	56	10	11	53	52	14	6	59	5	4	62	63	1	8	58
8	19	45	21	22	41	47	57	9	18	44	20	60	42	19	56
1	42	40	26	27	37	23	64	55	20	28	33	29	40	45	10
63	50	29	35	34	32	15	2	54	44	38	31	35	26	21	11
62	49	33	31	30	36	16	3	12	43	39	30	34	27	22	53
4	17	28	38	39	25	48	61	13	24	25	36	32	37	41	52
5	18	20	44	43	24	46	60	51	46	48	16	15	23	47	14
59	9	55	54	12	13	51	7	7	60	61	3	2	64	57	6

16x16 Cell Layout Details

- Concentric Magic Square
- Unit cell selection algorithm

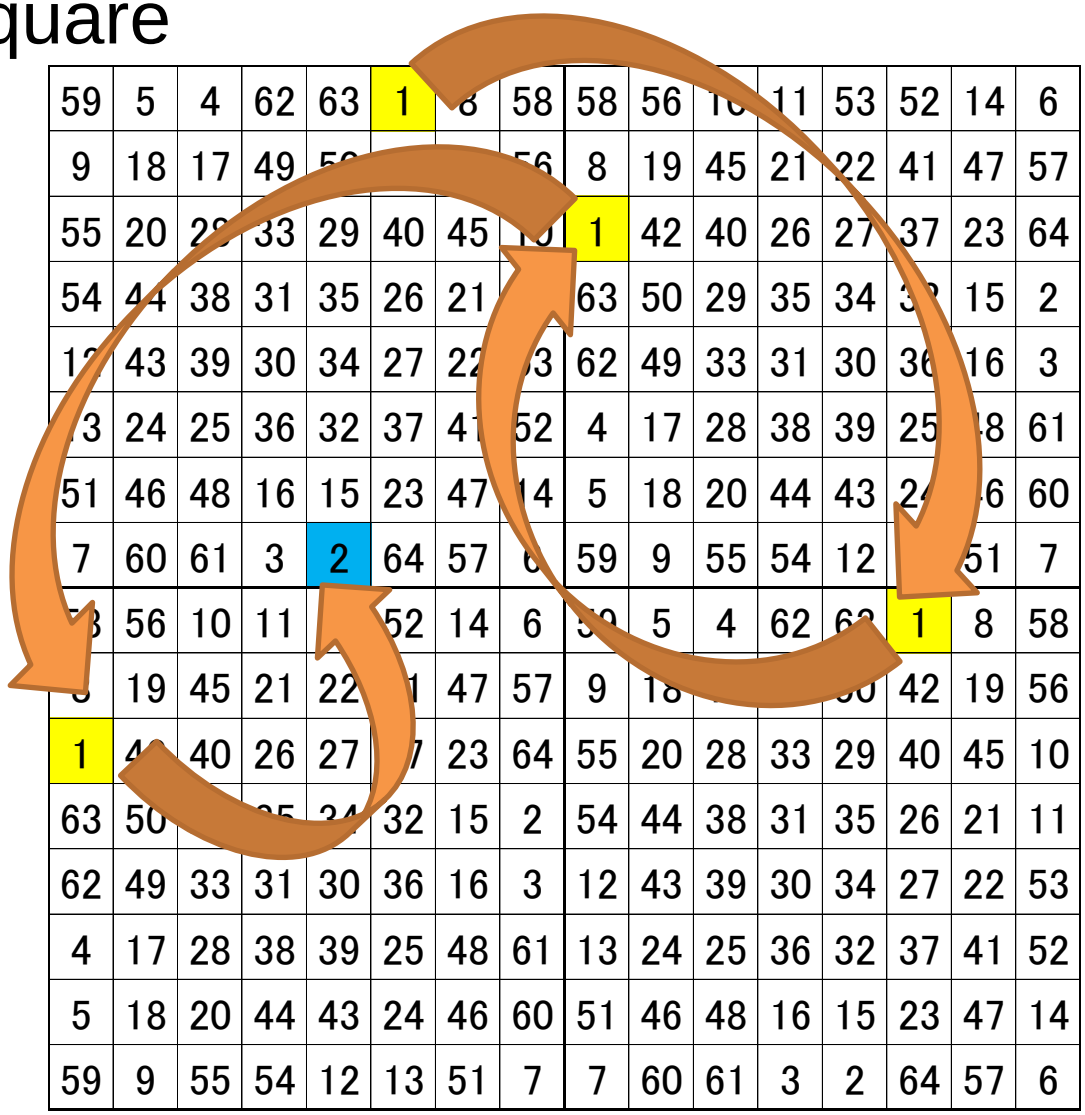
A1	B1
B2	A2



16x16 Cell Layout Details

- Concentric Magic Square
- Unit cell selection algorithm

A1	B1
B2	A2



16x16 Cell Layout Details

- Concentric Magic Square
- Unit cell selection algorithm

A1	B1
B2	A2

59	5	4	62	63	1	8	58	58	56	10	11	53	52	14	6
9	18	17	49	50	42	19	56	8	19	45	21	22	41	47	57
55	20	28	33	29	40	45	10	1	42	40	26	27	37	23	64
54	44	38	31	35	26	21	11	63	50	29	35	34	32	15	2
12	43	39	30	34	27	22	53	62	49	33	31	30	36	16	3
13	24	25	36	32	37	41	52	4	17	28	38	39	25	48	61
51	46	48	16	15	23	47	14	5	18	20	44	43	24	46	60
7	60	61	3	2	64	57	6	59	9	55	54	12	13	51	7
58	56	10	11	53	52	14	6	59	5	4	62	63	1	8	58
8	19	45	21	22	41	47	57	9	18	17	49	50	42	19	56
1	42	40	26	27	37	23	64	55	20	28	33	29	40	45	10
63	50	29	35	34	32	15	2	54	44	38	31	35	26	21	11
62	49	33	31	30	36	16	3	12	43	39	30	34	27	22	53
4	17	28	38	39	25	48	61	13	24	25	36	32	37	41	52
5	18	20	44	43	24	46	60	51	46	48	16	15	23	47	14
59	9	55	54	12	13	51	7	7	60	61	3	2	64	57	6

16x16 Cell Layout Details

- Concentric Magic Square
  - algorithm

1. 1 in A1
2. 1 in A2
3. 1 in B1
4. 1 in B2
5. 2 in A1

⋮

1023. 256 in B1
1024. 256 in B2

A1	B1
B2	A2

59	5	4	62	63	1	8	58	58	56	10	11	53	52	14	6
9	18	17	49	50	42	19	56	8	19	45	21	22	41	47	57
55	20	28	33	29	40	45	10	1	42	40	26	27	37	23	64
54	44	38	31	35	26	21	11	63	50	29	35	34	32	15	2
12	43	39	30	34	27	22	53	62	49	33	31	30	36	16	3
13	24	25	36	32	37	41	52	4	17	28	38	39	25	48	61
51	46	48	16	15	23	47	14	5	18	20	44	43	24	46	60
7	60	61	3	2	64	57	6	59	9	55	54	12	13	51	7
58	56	10	11	53	52	14	6	59	5	4	62	63	1	8	58
8	19	45	21	22	41	47	57	9	18	17	49	50	42	19	56
1	42	40	26	27	37	23	64	55	20	28	33	29	40	45	10
63	50	29	35	34	32	15	2	54	44	38	31	35	26	21	11
62	49	33	31	30	36	16	3	12	43	39	30	34	27	22	53
4	17	28	38	39	25	48	61	13	24	25	36	32	37	41	52
5	18	20	44	43	24	46	60	51	46	48	16	15	23	47	14
59	9	55	54	12	13	51	7	7	60	61	3	2	64	57	6

Represent pseudo-random switching while taking care of center and corners

Simulation Results (linear error case)

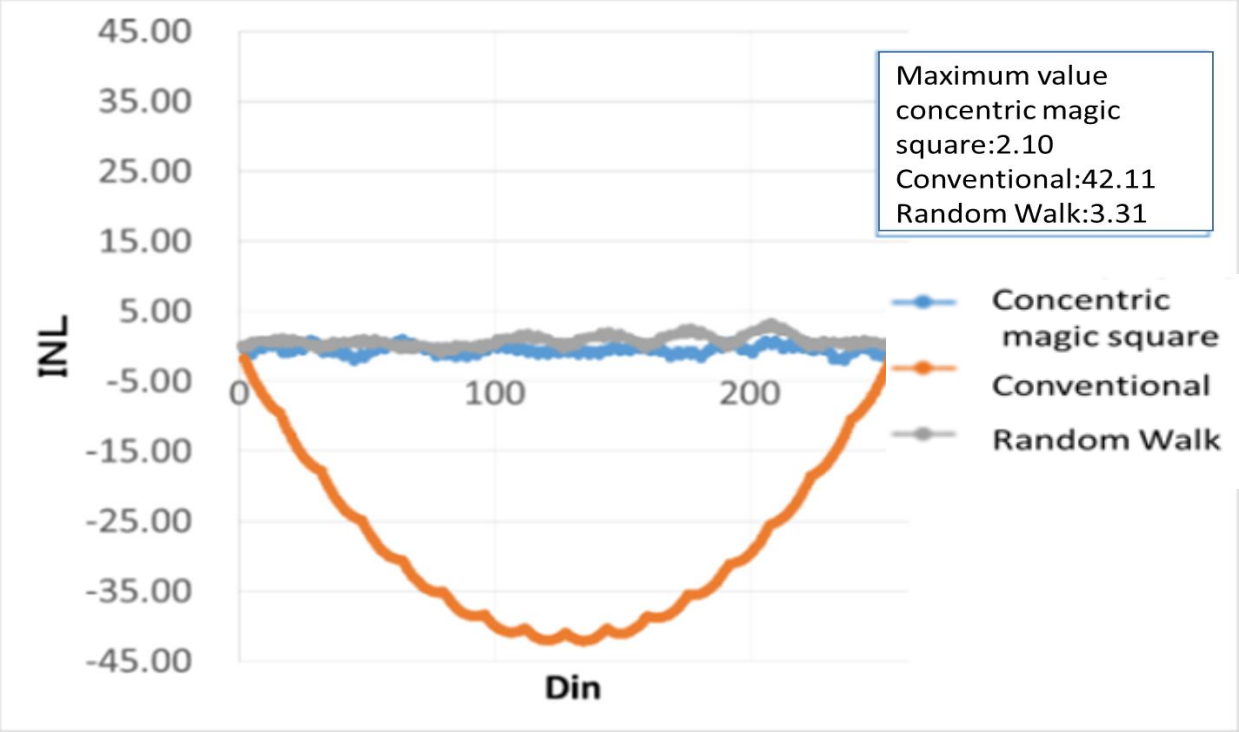
◆ Concentric Magic Square

✓ Linear Error (Current Cell Systematic Mismatch)

$$\varepsilon_l(x,y) = g_l * \cos\theta * x + g_l * \sin\theta * y$$

$\theta = 30°$

$g_l = 1$

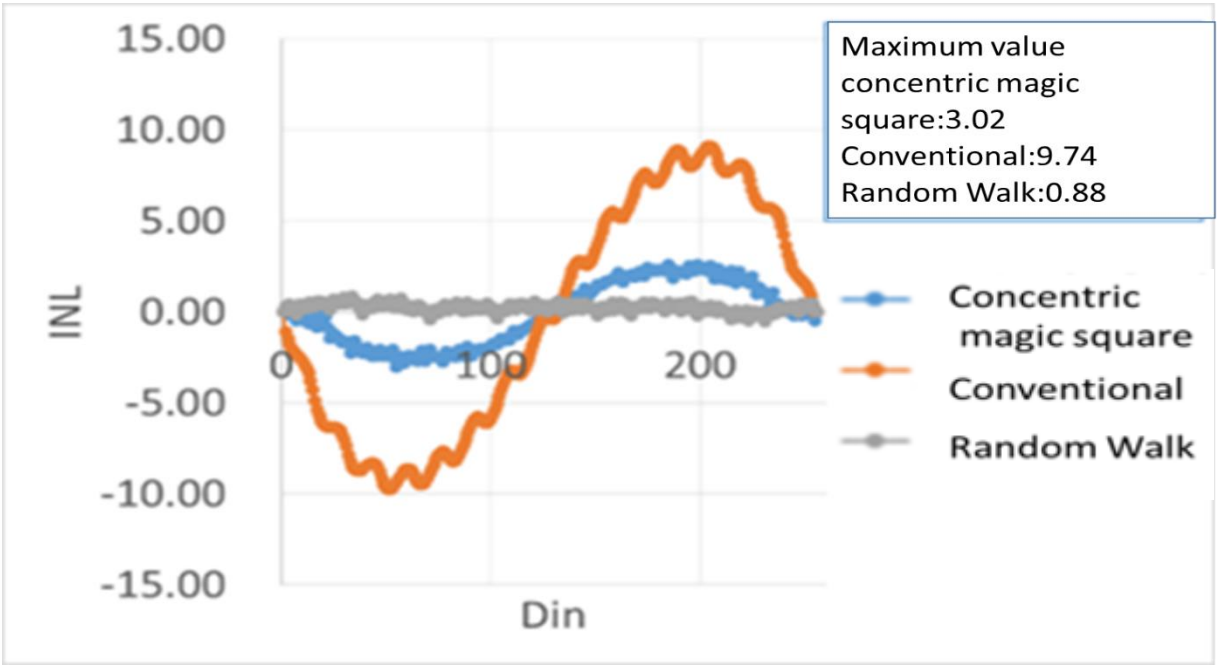


Simulation Results (quadratic error case)

- ◆ Concentric Magic Square
- ✓ Quadratic Error (Current Cell Systematic Mismatch)

$$\varepsilon_q(x,y) = g_q * (x^2 + y^2) - a_0$$

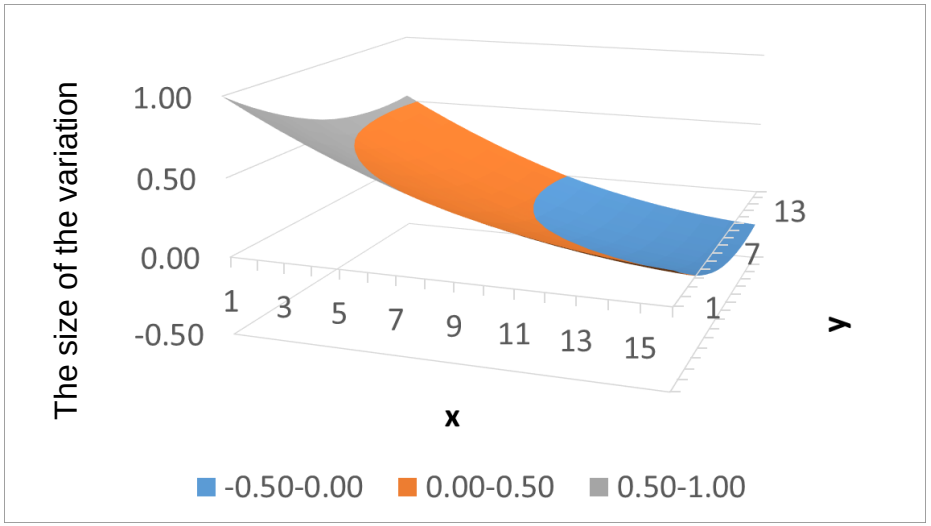
$$g_q = 1, a_0 = 0$$

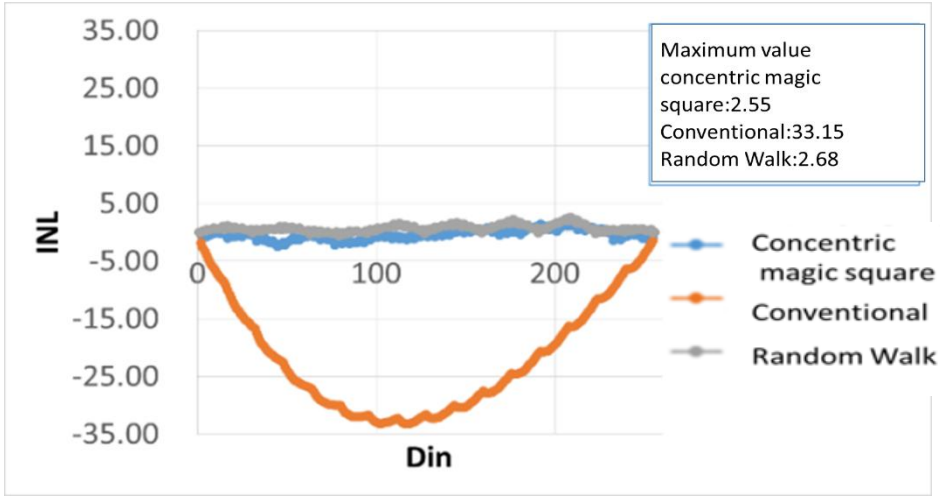


Simulation Results (joint error case)

- Concentric Magic Square
  - ✓ Joint Error

Linear > Quadratic case

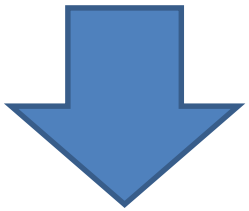




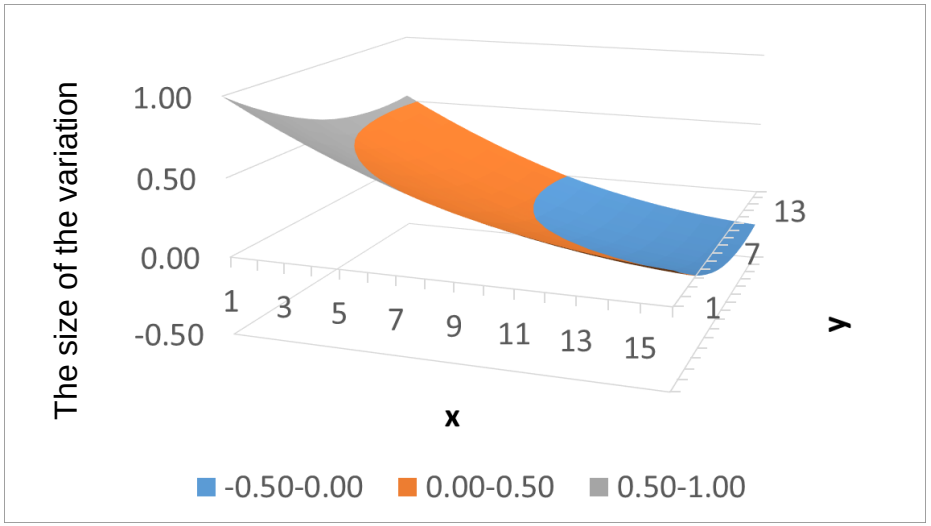
Simulation Results (joint error case)

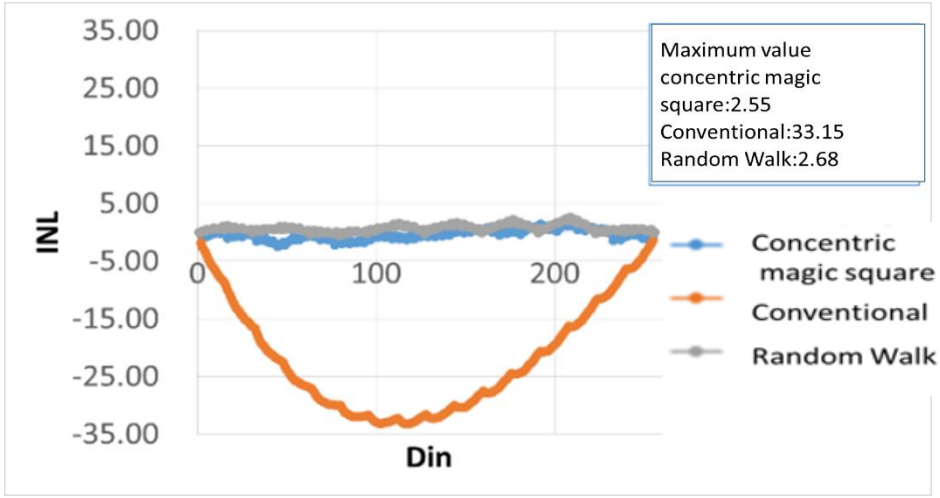
- Concentric Magic Square
  - ✓ Joint Error

Linear > Quadratic case



Magic square is better

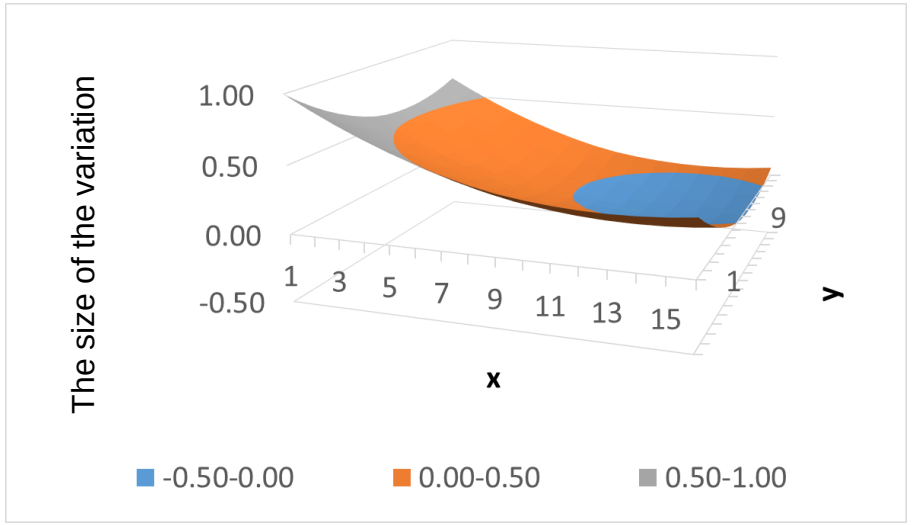


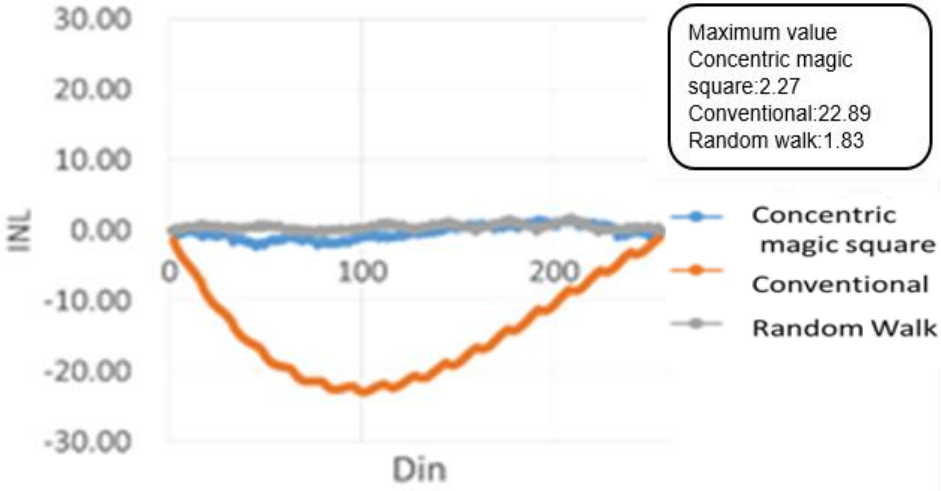


Simulation Results (joint error case)

- Concentric Magic Square
  - ✓ Joint Error

Linear < Quadratic case

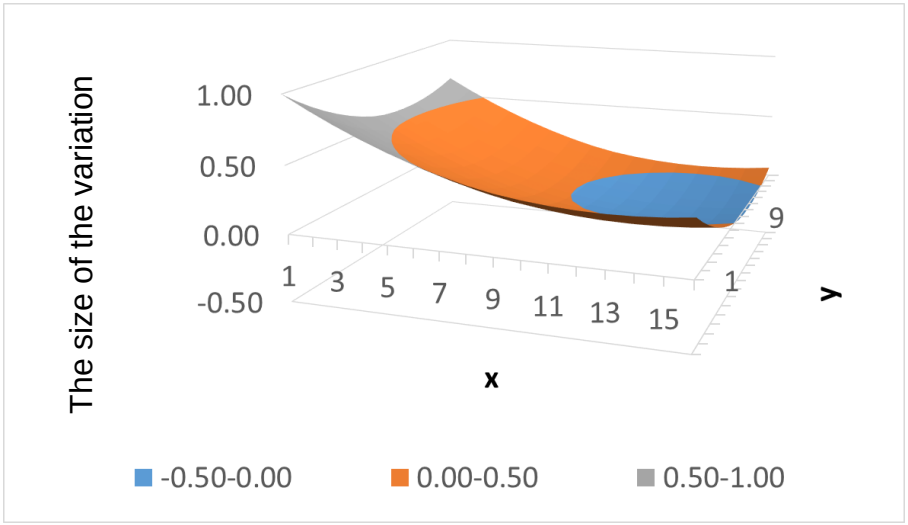




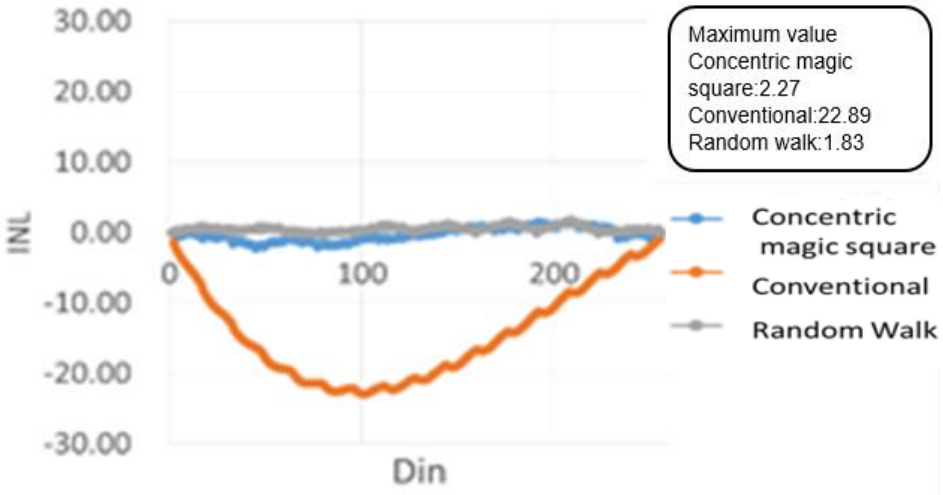
Simulation Results (joint error case)

- Concentric Magic Square
  - ✓ Joint Error

Linear < Quadratic case



Random Walk is better



Is the magic square suitable for temporary variation?

Contents

What is Latin Square ?

- $n \times n$ array filled with n different symbols
- Each symbols occurring exactly once in each row and column

Example :

A	B	C
C	A	B
B	C	A

3×3 Latin square

1	2	3	4
3	4	1	2
4	3	2	1
2	1	4	3

4×4 Latin square

Leonhard Euler(1707-1783)
Swiss mathematician, physicist

Latin Square for Layout Algorithm

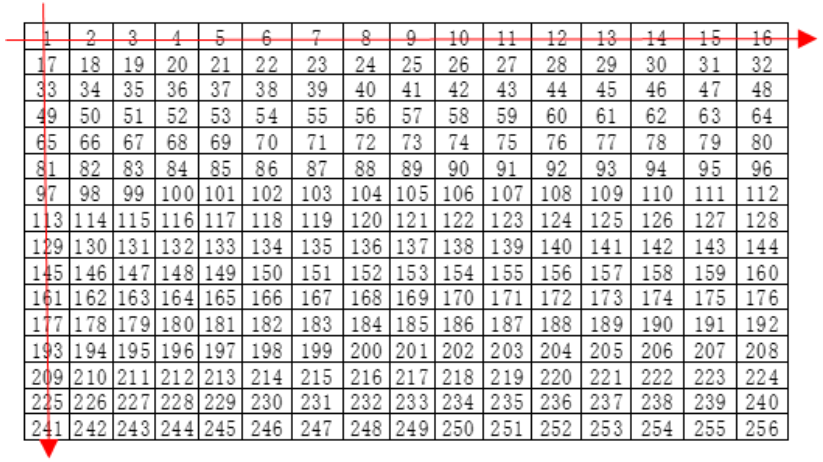
1	2	N	3	N-1	4	n-2	5	n-3	6	n-4	7	n-5	8	n-6	9

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

Considering a “complete Latin square”; for even n，
put the numbers I through n in the first row in the following order：
1, 2, n, 3, n-1,…., n/2+2, n/2+1.

Simulation Conditions

- 8-bit unary DAC
  - Static performance (INL)
  - Dynamic performance (SFDR)



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112
113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128
129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144
145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160
161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176
177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192
193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208
209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224
225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256

- Compared three methods
  - Complete Latin Square
  - Common Centroid
  - Unary Layout

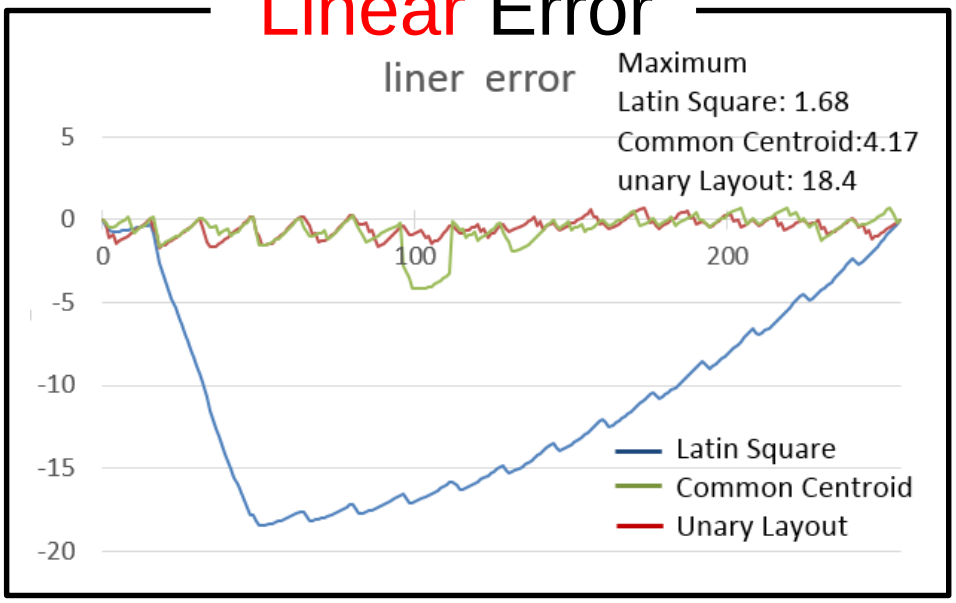
1	3	6	7	15	9	12	13	4	5	8	2	10	11	14	16
3	2	4	5	8	16	10	11	6	7	1	9	12	13	15	14
6	4	2	3	5	7	16	9	8	1	10	12	14	15	13	11
7	5	3	1	4	6	8	15	2	9	11	13	16	14	12	10
15	8	5	4	1	3	6	7	10	11	14	16	13	12	9	2
9	16	7	6	3	2	4	5	12	13	15	14	11	10	1	8
12	10	16	8	6	4	2	3	14	15	13	11	9	1	7	5
13	11	9	15	7	5	3	1	16	14	12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

- Mismatch of current sources
  - Current sources have average value of 1.0
  - Random number between -1 < mismatch < +1 (uniform distribution)

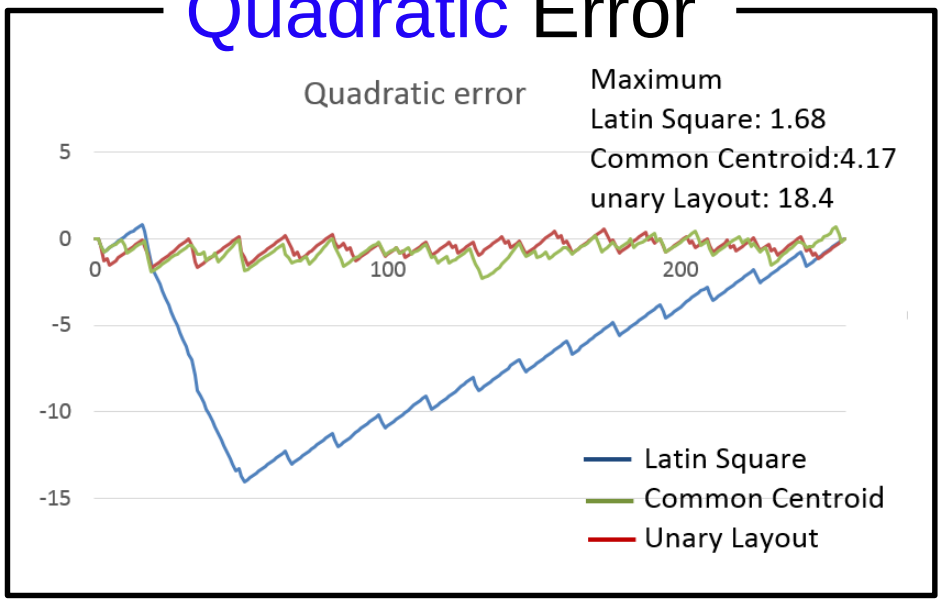
Simulation Results (INL)

- Standard Latin square layout algorithm

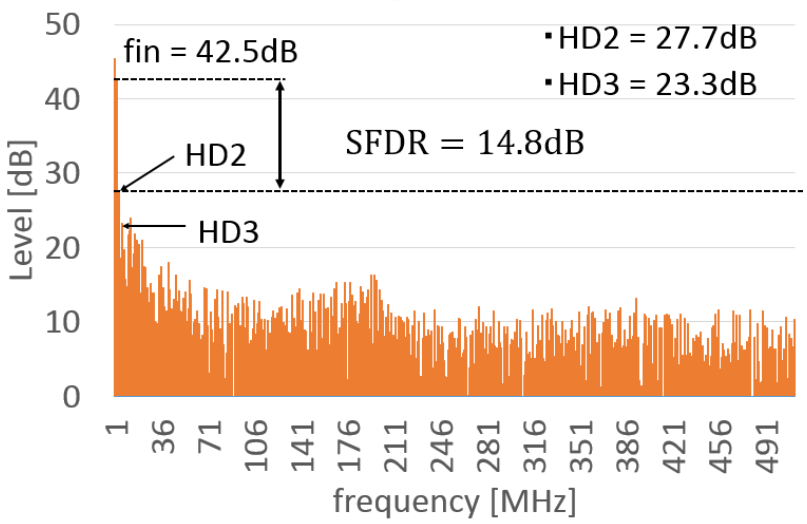
Linear Error



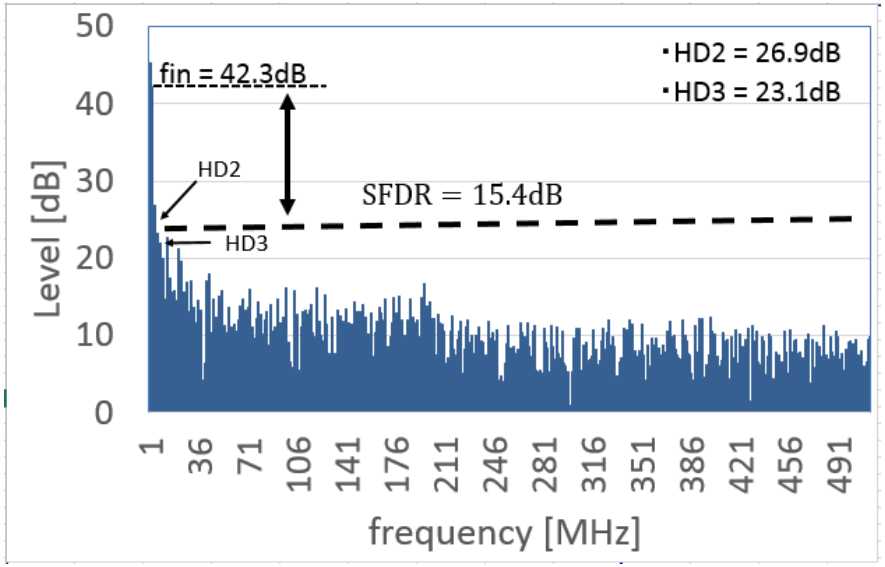
Quadratic Error



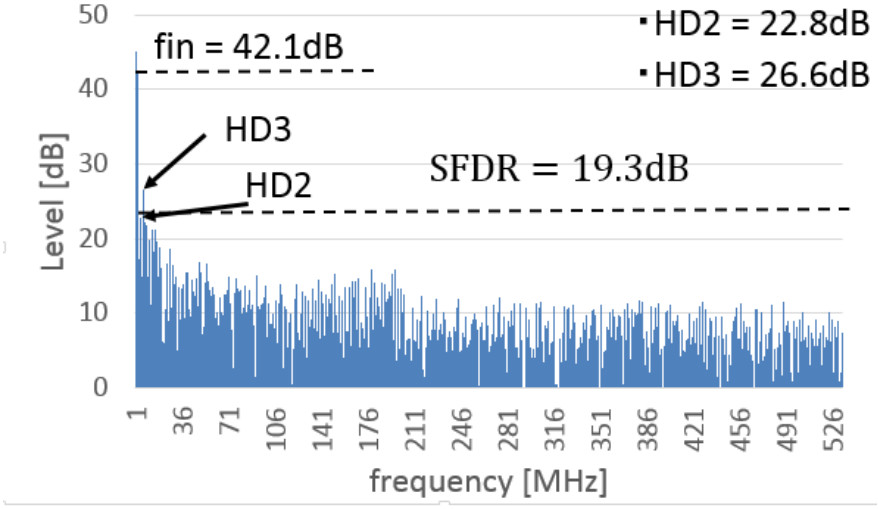
Simulation Results (SFDR)



Regular layout



Common centroid



Latin square

SFDR improved !

Summary

- Research Objective
- Segment Type DA Converter
- Characteristic of Variation in Circuit Element
- Proposed Method
  - Magic Square
  - Latin Square
- Conclusion

Conclusion

- Unary DAC linearity improvement
  - Unit current cell systematic mismatch effects cancellation
  - Unit current cell layout algorithm based on magic square and Latin square

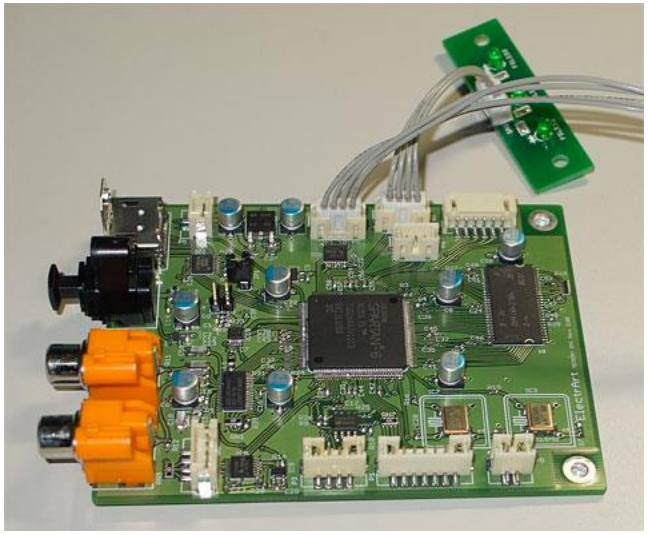
- Simulation validation
  - INL improvement
  - SFDR improvement

Final Statement

温故知新

Classical mathematics can contribute modern technology.

Leonhard Euler(1707～1783)



Thank you for listening

謝謝