Derivation of Digital-to-Analog Converter Architectures Based on Number Theory.

Xueyan Bai (t181d051@gunma-u.ac.jp)
Gunma University
Dan Yao
Gunma University
Yuanyang Du
Gunma University
Minh Tri Tran
Gunma University
Shogo Katayama
Gunma University
Jianglin Wei
Gunma University
Yujie Zhao
Gunma University
Anna Kuwana
Gunma University
Haruo Kobayashi
Gunma University
Kazuyoshi Kubo
National Institute of Technology, Oyama College

Research Article

Keywords: Digital-to-Analog Converter, DAC, Number Theory, Polygonal Number, Prime Number

Posted Date: May 17th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1603908/v1

Derivation of Digital-to-Analog Converter Architectures Based on Number Theory

Xueyan Bai[1] • Dan Yao[1] • Yuanyang Du[1] • Minh Tri Tran[1] • Shogo Katayama[1] • Jianglin Wei[1] • Yujie Zhao[1] • Anna Kuwana[1] • Haruo Kobayashi[1] • Kazuyoshi Kubo[2]

Abstract

This paper investigates possibilities of deriving new digital-to-analog converter (DAC) architectures based on polygonal and prime number theories. As the result, configurations of polygonal number DACs and prime number DACs are obtained; each consists of a few current sources, a resistor network, switch arrays and a decoder circuit. Whole circuits are designed and their operations are confirmed with simulation; they work as DAC in principle. In many cases, analog/mixed-signal circuit architecture design relies on intuitions and experiences of the designer, but here we demonstrate that it is feasible to derive different DAC architectures from conventional ones based on mathematical theory.

Keywords Digital-to-Analog Converter, DAC, Number Theory, Polygonal Number, Prime Number

1 Introduction

Analog/mixed-signal circuit design is art rather than technology, with which industry can differentiate their electronic products. There analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) are especially important [1-4]. Their design mostly relies on intuitions and experiences of mature designers, rather than mathematical theory; only one exception is analog filter synthesis and analysis [5]. On the contrary, the authors have been involved in the research for applications of classical mathematics - such as number theory [6] - to their design; in this viewpoint, our research results have been reviewed and summarized in [7-13]. This paper introduces new DAC architecture derivation from number theory to validate our argument that classical mathematics can explore new analog and mixed-signal circuit design. The derived DAC architectures may not well incorporate circuit non-ideality effects such as device mismatches. However, this attempt may lead to new DAC architecture derivation methodology, and this paper shows its first step.

Our previous research results for applications of classical mathematics to analog/mixed-signal circuit design are summarized as follows.

Fibonacci sequence weight SAR ADC: We have investigated SAR ADC design using a redundant SAR search algorithm with Fibonacci sequence weight. We showed that this method can realize high speed SAR AD conversion when the internal DAC incomplete settling is considered [14].

Metallic Ratio Sampling: We have investigated efficient waveform acquisition conditions between the measured waveform repetitive frequency (f_{sig}) and the sampling clock frequency (f_{CLK}) in an equivalent-time sampling system, when the measured waveform is periodic. We have obtained that in case that $f_{\mathrm{CLK}}/f_{\mathrm{sig}}$ is a metallic ratio, waveform missing phenomena for the equivalent-time sampling can be avoided and highly efficient waveform acquisition sampling can be realized [15-17]. This technique can be used for analog/mixed-signal IC testing where the input signal is controllable.

Residue Sampling: We have investigated the residue sampling circuit which provides high-frequency signal estimation using multiple low-frequency sampling circuits following an analog Hilbert filter and ADCs; the sampling frequencies are relatively prime. It is based on aliasing phenomena in the frequency domain for waveform sampling and the residue number theory [18, 19].

Efficient ADC Histogram Testing Condition: We have studied the ADC testing efficiency improvement of the histogram method by investigating the ratio between the input frequency and the sampling frequency to shorten ADC test time, and we have found that the metallic ratio is effective [20].

Non-uniform Current Division Resistive DAC: We have studied design and analysis of DACs based on the non-uniform current division resistive ladder, and proposed a new configuration DAC with segmentation of binary, quaternary and unary resistive-ladders, which enables two times gain with equivalent chip area and current sources to the conventional one [21-23].

Gray Code Input DAC: We have also investigated three types of Gray-code input DAC architectures (current-steering, charge-mode and voltage-mode DACs) for glitch reduction and hence clean signal can be generated [24, 25].

2D Layout of Unit Cells with Pseudo Random Selection Order for Unary DAC: We have investigated pseudo random selection order algorithms for the segmented DAC linearity improvement, by cancelling systematic mismatch

Xueyan Bai
t202d002@gunma-u.ac.jp
Anna Kuwana
kuwana.anna@gunma-u.ac.jp
Haruo Kobayashi
koba@gunma-u.ac.jp

1. Division of Electronics and Informatics, Faculty of Science and Technology, Gunma University, 1-5-1 Tenjin-cho, Kiryu-shi, Gunma, 376-8515, Japan
2. National Institute of Technology (KOSEN), Oyama College, 771 Nakakuki, Oaza, Oyama, Tochigi 323-0806, Japan

effects among unit cells [26-29]. We consider their 2D array layout with systematic errors. If they are laid out and selected in a regular manner, the systematic errors are accumulated at the DAC output, which causes large non-linearity. If they are selected pseudo-randomly, they can be cancelled to some extent. The investigated pseudo-random selection algorithms are based on Magic Square, Latin Square as well as Euler's Knight Tour.

Polygonal Number DAC and Prime Number DAC: We have proposed the preliminary ideas of the following DAC configurations based on number theory [30-32]: (i) The DAC consists of N current sources, N switch arrays, an N-polygonal number weighted resistor network, and a decoder (N= 3, 4, 5, ...); this is based on the polygonal number theory. (ii) The DAC consists of 2 current sources, 2 switch arrays, a prime number weighted resistor network, and a decoder (N= 3, 4, 5, ...); this is based on the Goldbach conjecture. This paper discusses the derivation of these DACs from the number theory in details. This paper shows that the derivation of very new DAC configurations from the number theory is possible, and demonstrates their operations in principle as the first step. Also the possibility of dynamic element matching (DEM) technique usage [3, 12] to take care of the device mismatch effects is described.

The outline of this paper is as follows: Section 2 shows the derivation of the polygonal number DAC from the polygonal number theorem and its configuration as well as operation verification with simulation. Section 3 shows the derivation of the prime number DAC from Goldbach conjecture and its configuration as well as operation with simulation. Section 4 provides conclusion.

2 Polygonal Number DAC

This section describes derivation of our polygonal number DACs.

2.1 Polygonal Number Theory

This subsection briefly explains polygonal numbers in number theory. We take the triangular number as an example, and infer that this consideration can also applicable to other polygonal numbers through the simulation results of triangular number.

Polygonal numbers are represented as dots or pebbles arranged in the shape of a regular polygon; they are triangular numbers, square numbers, pentagonal numbers, hexagonal numbers, heptagonal numbers, octagonal numbers, and so on, as shown in Fig. 1.

For example, the triangular numbers are given by as follows (see Fig. 1 (a)):1, 3, 6, 10, 15, 21, 28, 36, 45, 55, 66, 78, 91, 105…, n(n+1)/2, …

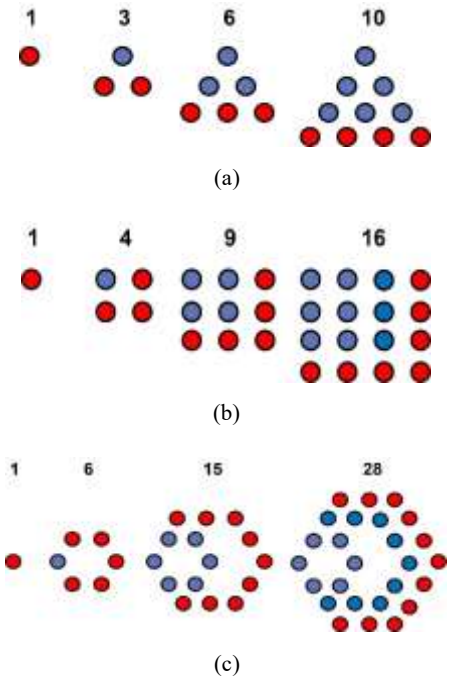


Fig. 1 Explanation of polygonal numbers. (a) Triangular numbers. (b) Square numbers. (c) Hexagonal numbers.

We design DACs based on the following triangular number theoretical properties.

Triangular number theorem:

"Any natural number is composed of 3 or less than 3 triangular numbers".

Fig. 2 shows its explanation.

1 :	1	16 :	1+15
2 :	1+1	17 :	1+1+15
3 :	3	18 :	3+15
4 :	1+3	19 :	1+3+15
5 :	1+1+3	20 :	10+10
6 :	6	21 :	21
7 :	1+6	22 :	1+21
8 :	1+1+6	23 :	1+1+21
9 :	3+6	24 :	3+21
10 :	10	25 :	1+3+21
11 :	1+10	26 :	1+10+15
12 :	1+1+10	27 :	6+21
13 :	3+10	28 :	28
14 :	1+3+10	29 :	1+28
15 :	15	30 :	1+1+28

Fig. 2 Explanation of triangular number theorem.

2.2 Triangular Number DAC

Our derived DAC based on the triangular number theorem is shown in Fig. 3. It consists of 3 current sources, 3 switch arrays, a triangular number weighted resistor network and a decoder circuit.

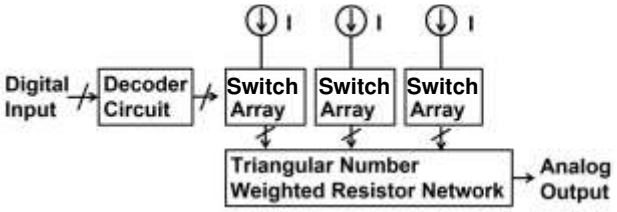


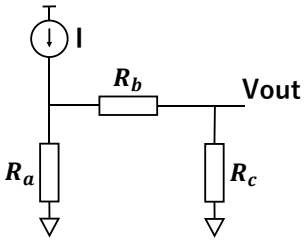
Fig. 3. Proposed triangular number DAC.

Triangular Number Weighted Resistor Network:

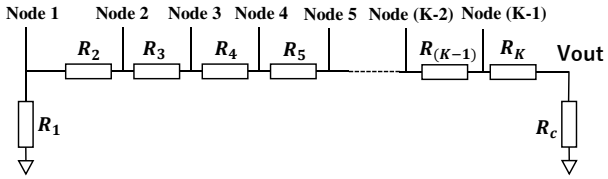
First, we consider the resistor network in Fig. 4 (a) and we obtain the output voltage V_{OUT} as follows:

$$V_{OUT} = I\, R_a R_c/(R_a + R_b + R_c). \quad (1)$$

Here R_c is a load resistor and we see that it changes the gain of the resistive network. Its extension to the network of K resistors and a load resistor R_c is shown in Fig. 4 (b).



(a)



(b)

Fig. 4 Basic resistor networks. (a) 3 resistors with an input current source. (b) Extension to K resistors and a load resistor R_c

The key component of the triangular number DAC is the triangular number weighted resistor network in Fig. 5, where R_n in Fig. 4 (b) is replaced with n R (n=1, 2,…, K) and R is a unit resistor.

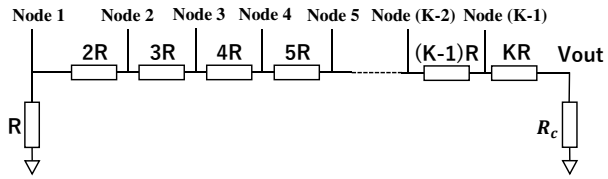


Fig. 5 Triangular number weighted resistor network.

Fig. 6 shows the triangular number weighted resistor network with K=5 and an input current source applied to each node.

We have the following from Fig. 4 (a) and Eq. (1):

$R_a + R_b = R + 2R + 3R + 4R + 5R = 15R$

$V_{OUT} = I\, R_a\, [R_c\, /\, (R_a + R_b + R_c)]$.

Also $R_c = R$, and then $V_{OUT} = \left(\frac{1}{16}\right) R_a I$.

In Fig. 6 (a), $R_a = R$, $R_b = 2R + 3R + 4R + 5R = 14R$ and $V_{OUT} = \left(\frac{1}{16}\right) RI$.

In Fig. 6 (b), $R_a = R + 2R = 3R$, $R_b = 3R + 4R + 5R = 3R$ and $V_{OUT} = \left(\frac{3}{16}\right) RI$.

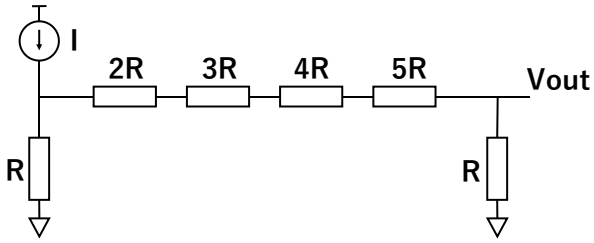
In Fig. 6 (c), $R_a = R + 2R + 3R = 5R$, $R_b = 4R + 5R = 9R$ and $V_{OUT} = \left(\frac{6}{16}\right) RI$.

In Fig. 6 (d), $R_a = R + 2R + 3R + 4R = 10R$, $R_b = 5R$ and $V_{OUT} = \left(\frac{10}{16}\right) RI$.

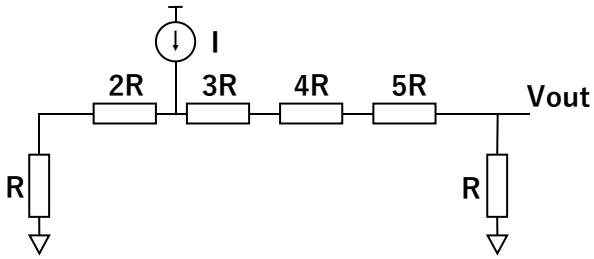
In Fig. 6 (e), $R_a = R + 2R + 3R + 4R + 5R = 15R$, $R_b = 0$ and $V_{OUT} = \left(\frac{15}{16}\right) RI$.

Notice that 1, 3, 6, 10, 15 are triangular numbers.

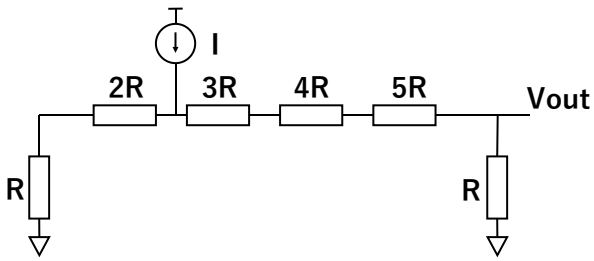
We see that the triangular number weighted resistor network generates a triangular number weighted voltage at V_{OUT} when a current is injected to one node.



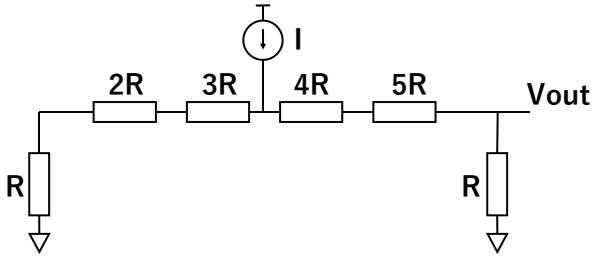
(a)



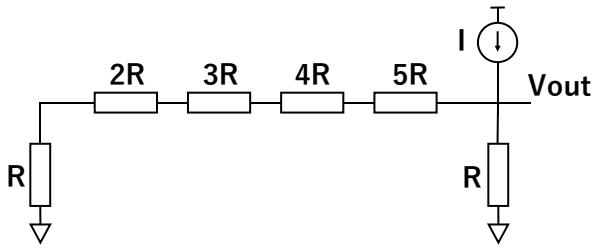
(b)



(c)



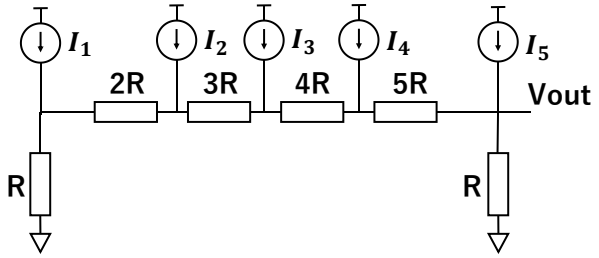
(d)



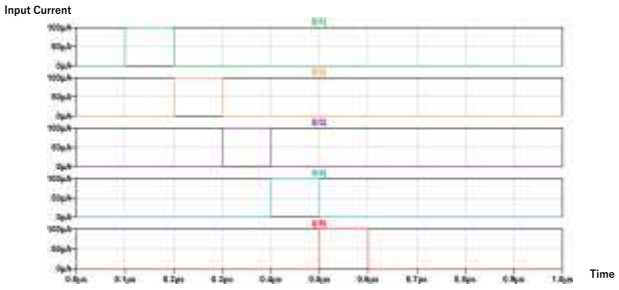
(e)

Fig. 6 Operation of the triangular number weighted resistor network, which generates a triangular number weighted voltage at V_{OUT}. (a) V_{OUT} = (1/16) RI. (b) V_{OUT}= (3/16) RI. (c) V_{OUT}= (6/16) RI. (d) V_{OUT}= (10/16) RI. (e) V_{OUT} = (15/16) RI.

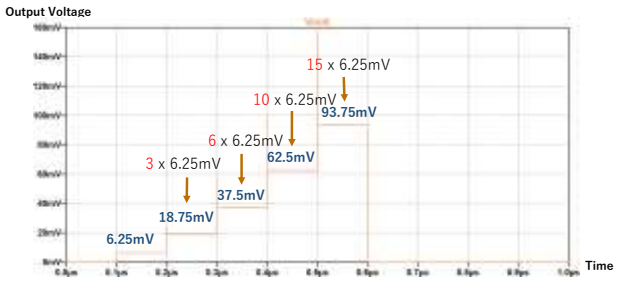
We have performed SPICE simulation for the circuits in Fig. 7. Simulation circuit is shown in Fig. 7 (a), where R = 1kΩ. The waveforms of the input current sources I_1 ~I_5 are shown in Fig. 7 (b) and the waveform of Vout is shown in Fig. 7 (c); we see that this simulation result confirms the operations of the circuits in Fig. 6, with R = 1kΩ and I = 100μA.



(a)



(b)

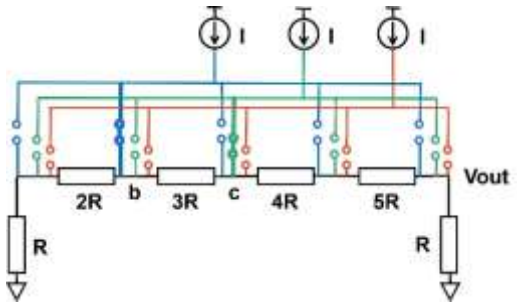


(c)

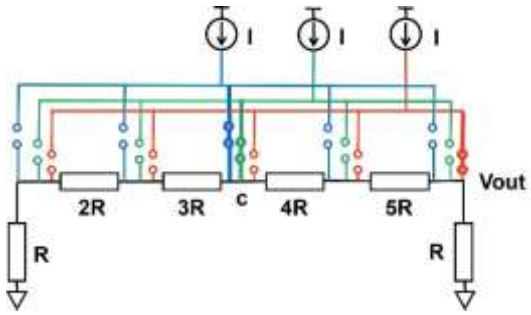
Fig. 7 SPICE simulation of the proposed triangular number weighted resistor network. (a) Simulation circuit. (b) Input current source waveforms. (c) Output voltage (V_{OUT}) waveform.

Three Current Sources and Three Switch Arrays:

Based on the theoretical properties of the triangular numbers and the superposition principle, we obtain the input combination for the digital input from 0 to 30 by controlling three switch arrays for three current sources, according to Fig. 2. The output of the three current sources is the superposition of the output results of a single current source, so we obtain the results as shown in Fig. 7. For example, in the case of the input 9, we need to input two current sources, that is, superimposition of Fig. 6 (b) and Fig. 6 (c) becomes Fig. 8 (a).



(a)



(b)

Fig. 8 Superposition of three current sources to the investigated triangular number weighted resistor network. (a) In case that the digital input is 9. $V_{out} = [(3+6)/16]RI$. (b) In case that the digital input is 27. $V_{out} = [(6+6+15)/16]RI$

Decoder Design:

We have designed a decoder logic for a 6-bit triangular number DAC. Its digital inputs are from 0 to 63, and three switch arrays corresponding to three current sources is represented by Sa, Sb and Sc respectively. There are 11 switches in the switch array of each current source, which are represented by Sa00... Sa10, Sb00... Sb10, Sc00... Sc10.

Due to the large number, we cannot verify the assumptions one by one, so we write an automatic verification program to detect whether all the input conditions are in line with our conjecture. The following is the design principle of the verification program:

Suppose that the digital input is 5 in decimal or B5, B4, B3, B2, B1 and B0 are 000101 in binary. Notice that 1+1+3=5 as shown in Fig. 2. Then three switches of Sa1, Sb1, and Sc2 are ON (Sa1 =1, Sb1 = 1, Sc2 = 1) and the other switches are OFF, as shown in Figs. 9, 10.

The triangular number DAC decoder has the digital inputs of B5, B4, B3, B2, B1, B0 (from 0 to 63 in decimal) and the digital outputs of Sa00, Sa01, Sa02, … Sa10, Sb00, …Sb10, Sc00, … Sc10. Each output logical expression from B5, B4, B3, B2, B1, B0 can be obtained from Fig. 2, and each logical expression is verified by a C program with the following algorithm:

Data A, Data B, Data C and Data are defined as follows:

$$\begin{aligned}\text{DataA} = {} & 0\cdot\text{Sa00}+1\cdot\text{Sa01}+3\cdot\text{Sa02}+6\cdot\text{Sa03}\\ & +10\cdot\text{Sa04}+15\cdot\text{Sa05}+21\cdot\text{Sa06}\\ & +28\cdot\text{Sa07}+36\cdot\text{Sa08}+45\cdot\text{Sa09}\\ & +66\cdot\text{Sa10}.\end{aligned}$$

$$\begin{aligned}\text{DataB} = {} & 0\cdot\text{Sb00}+1\cdot\text{Sb01}+3\cdot\text{Sb02}+6\cdot\text{Sb03}\\ & +10\cdot\text{Sb04}+15\cdot\text{Sb05}+21\cdot\text{Sb06}\\ & +28\cdot\text{Sb07}+36\cdot\text{Sb08}+45\cdot\text{Sb09}\\ & +66\cdot\text{Sb10}.\end{aligned}$$

$$\begin{aligned}\text{DataC} = {} & 0\cdot\text{Sc00}+1\cdot\text{Sbc01}+3\cdot\text{Sbc02}+6\cdot\text{Sbc03}\\ & +10\cdot\text{Sbc04}+15\cdot\text{Sc05}+21\cdot\text{Sc06}\\ & +28\cdot\text{Sc07}+36\cdot\text{Sc08}+45\cdot\text{Sc09}\\ & +66\cdot\text{Sc10}.\end{aligned}$$

$$\begin{aligned}\text{Data} = {} & \text{B0}+2\cdot\text{B1}+4\cdot\text{B2}+8\cdot\text{B3}+16\cdot\text{B4}\\ & +32\cdot\text{B5}.\end{aligned}$$

Here Sa00, Sa01, Sa02, … Sa10, Sb00, …Sb10, Sc00, … Sc10 are written as designed decoder logical expressions of B5, B4, B3, B2, B1, B0.

The program runs from B5 B4 B3 B2 B1 B0 = 000000 to 11111 and verified the following in all cases:

If $\text{Data} = \text{DataA} + \text{DataB} + \text{DataC}$, the output is OK.

If $\text{Data} != \text{DataA} + \text{DataB} + \text{DataC}$, the output is WRONG.

When for all B5, B4, B3, B2, B1, B0, the output is OK, then the decoder logical expressions are correct. The verification program execution results are as shown as Fig. 11.

B5	B4	B3	B2	B1	B0
0	0	0	1	0	1

Sa0	Sa1	Sa2	Sa3	Sa4	Sa5	Sa6	Sa7	Sa8	Sa9	Sa10
0	1	0	0	0	0	0	0	0	0	0
Sb0	Sb1	Sb2	Sb3	Sb4	Sb5	Sb6	Sb7	Sb8	Sb9	Sb10
0	1	0	0	0	0	0	0	0	0	0
Sc0	Sc1	Sc2	Sc3	Sc4	Sc5	Sc6	Sc7	Sc8	Sc9	Sc10
0	0	1	0	0	0	0	0	0	0	0

Fig. 9 DAC decoder logic for the digital input of 5.

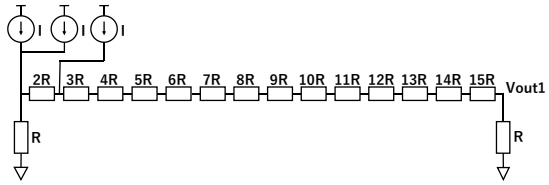


Fig. 10 Three current sources connection to the triangular number weighted network for the digital input of 5.

```
B5=0, B4=0, B3=0, B2=0, B1=0, B0=0  OK  0=0+0+0
B5=0, B4=0, B3=0, B2=0, B1=0, B0=1  OK  1=1+0+0
B5=0, B4=0, B3=0, B2=0, B1=1, B0=0  OK  2=1+1+0
B5=0, B4=0, B3=0, B2=0, B1=1, B0=1  OK  3=3+0+0
B5=0, B4=0, B3=0, B2=1, B1=0, B0=0  OK  4=1+3+0
B5=0, B4=0, B3=0, B2=1, B1=0, B0=1  OK  5=1+1+3
B5=0, B4=0, B3=0, B2=1, B1=1, B0=0  OK  6=6+0+0
B5=0, B4=0, B3=0, B2=1, B1=1, B0=1  OK  7=1+6+0
B5=0, B4=0, B3=1, B2=0, B1=0, B0=0  OK  8=1+1+6
B5=0, B4=0, B3=1, B2=0, B1=0, B0=1  OK  9=3+6+0
B5=0, B4=0, B3=1, B2=0, B1=1, B0=0  OK  10=10+0+0
```

Fig. 11 Execution result of the triangular number DAC decoder verification program.

Remark 1: According to the triangular number theory, any natural number is the sum of three, two or one of triangular numbers. However, notice that its expression is not unique; a given natural number n can be represented by $n = n_1 + n_2 + n_3$ and $n = m_1 + m_2 + m_3$ where n_1, n_2, n_3, m_1, m_2 and m_3 are triangular numbers or 0. For example, in case n=29, 25=10+15 and 25=1+3+21. For the decoder design, we choose one of them for a given input data.

2.3 General Polygonal Number DAC

In general, N-polygonal number theorem is given as follows (N=3, 4, 5, 6, …):

The polygonal number theory:

"Any natural number is composed of N or less than N-polygonal numbers."

This polygonal number theorem was conjectured by French mathematician, Pierre de Fermat and proved by French mathematician, Augustin-Louis Cauchy.

Fig. 12 shows out proposed DAC configuration using N-polygonal number properties, which is composed of N current sources, N switch arrays, an N-polygonal number weighted resistor network and a decoder circuit; these are based on the above polygonal number theorem.

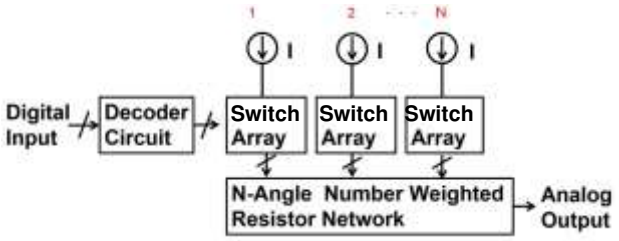


Fig. 12 DAC configuration based on N-polygonal number.

N-Polygonal Number Weighted Resistor Network:

Now we will explain N-polygonal number weighted resistor networks.

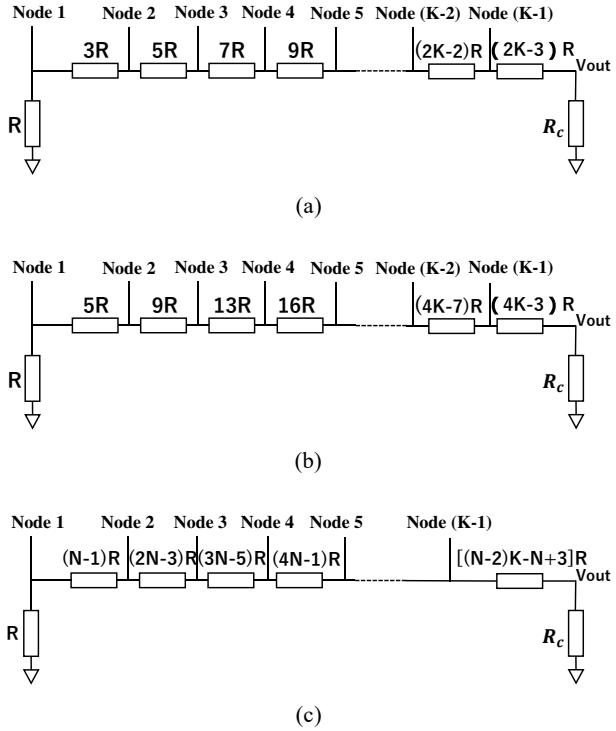


Fig. 13 Designed N-polygonal number weighted resistor network. (a) Square number weighted resistor network (N=4). (b) Hexagonal number weighted resistor network (N=6). (c) N-polygonal number weighted resistor network.

The n-th N-polygonal number $P_N\ (n)$ is given as follows:

$$P_N\ (n) = \frac{(n^2-n)N}{2} - n^2 + 2n \quad \text{(n=1, 2, 3, 4, 5, ….)}$$

or 1, N, 3N-3, 6N-8, 10N-15, ….

Then the N-polygonal number resistor network using Fig. 4 (b) is designed as follows:

R1=R

R2= (N-1) R

R3= (3N-3) R – (R1+R2) = (3N-3) R – (1+(N-1)) R

=(2N-3) R

R4 = (6N-8) R – (R1+R2+R3) R

= (6N-8) R – [1+(N-2) + (2N-3)] R

= (3N-4) R

R5 = (10N-15) R – (R1+R2+R3+R4) R

= (10N-15) R – [1+(N-2) + (2N-3) + (3N-4)] R

= (4N-7) R

Fig. 13 shows the N-polygonal number weighted resistor network with K=5 and an input current source is applied to each node.

$$R_a + R_b = R + (N-1)R + (2N-3)R + (3N-5) + (4N-7)R = (10N-15)\,R$$

$$V_{OUT} = I\,R_a\,[R_c\,/\,(R_a + R_b + R_c)].$$

Also $R_c = R$, and then $V_{OUT} = \left(\frac{1}{10N-14}\right) R_a I$.

In Fig. 13 (a), $R_a = R$, and $V_{OUT1} = \left(\frac{1}{10N-14}\right) RI$.

In Fig. 13 (b), $R_a = R + (N-1)R = NR$,

and $V_{OUT3} = \left(\frac{N}{10N-14}\right) RI$.

In Fig. 13 (c), $R_a = R + (N-1)R + (2N-3)R = (3N-3)\,R$, and $V_{OUT6} = \left(\frac{3N-3}{10N-14}\right) RI$.

In Fig. 13 (d), $R_a = R + (N-1)R + (2N-3)R + (3N-5)R = (6N-9)\,R$, and $V_{OUT10} = \left(\frac{6N-9}{10N-14}\right) RI$.

In Fig. 13 (e), $R_a = R + (N-1)R + (2N-3)R + (3N-5)R + (4N-7)R = (10N-15)\,R$,

and $V_{OUT15} = \left(\frac{10N-15}{10N-14}\right) RI$.

Notice that 1, N, 3N-3, 6N-9, 10N-15 are N-polygonal numbers. They are 1, 3, 6, 9, 15 for N=3 and they are 1, 4, 9, 25, 36 for N=4, while they are 1, 6, 15, 27, 45 for N=6.

We see that the N-polygonal number weighted resistor network generates an N-polygonal number weighted voltage at Vout when a current is injected to one node.

3 Prime Number DAC

This section describes derivation of our prime number DACs.

3.1 Prime Number Theory

Prime number is a natural number greater than 1 that cannot be formed by multiplying two smaller natural numbers.

Prime numbers：2, 3, 5, 7, 11, 13, 17, 19, 23, 29, ……

Goldbach conjecture is given as follows (Fig. 14):

"*All even numbers can be represented by the sum of two prime numbers.*"

2 :	2	32 :	13+19
4 :	2+2	34 :	17+17
6 :	3+3	36 :	17+19
8 :	3+5	38 :	19+19
10 :	3+7	40 :	17+23
12 :	5+7	42 :	19+23
14 :	7+7	44 :	13+31
16 :	5+11	46 :	23+23
18 :	7+11	48 :	19+29
20 :	7+13	50 :	19+31
22 :	11+11	52 :	23+29
24 :	11+13	54 :	23+31
26 :	13+13	56 :	19+37
28 :	11+17	58 :	29+29
30 :	13+17	60 :	29+31

Fig. 14 Explanation of Goldbach conjecture

The above conjecture has not been proved yet. However, we can check with numerical calculation that the above conjecture is valid, for example, up-to 2^{21} for 20-bit DAC design.

3.2 Prime Number DAC

Fig. 15 shows our proposed prime number DAC circuit based on Goldbach conjecture, which consists of 2 current sources, 2 switch arrays, a prime number weighted resistor network and a decoder circuit.

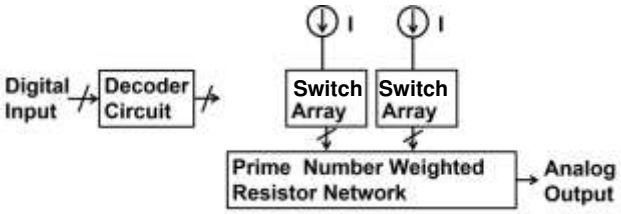


Fig. 15 Proposed prime number DAC.

We consider the mapping from even number obtained by addition of two prime numbers to the DAC digital input, as shown in Fig. 16. Then we have the digital input by "divide by 2 (1-bit right shift)" of two-prime-number addition.

Digital Input		Digital Input	
1 ← 2 :	2	16 ← 32 :	13+19
2 ← 4 :	2+2	17 ← 34 :	17+17
3 ← 6 :	3+3	18 ← 36 :	17+19
4 ← 8 :	3+5	19 ← 38 :	19+19
5 ← 10 :	3+7	20 ← 40 :	17+23
6 ← 12 :	5+7	21 ← 42 :	19+23
7 ← 14 :	7+7	22 ← 44 :	13+31
8 ← 16 :	5+11	23 ← 46 :	23+23
9 ← 18 :	7+11	24 ← 48 :	19+29
10 ← 20 :	7+13	25 ← 50 :	19+31
11 ← 22 :	11+11	26 ← 52 :	23+29
12 ← 24 :	11+13	27 ← 54 :	23+31
13 ← 26 :	13+13	28 ← 56 :	19+37
14 ← 28 :	11+17	29 ← 58 :	29+29
15 ← 30 :	13+17	30 ← 60 :	29+31

Fig 16 Mapping from even number obtained by addition of two prime numbers to digital input.

Prime Number Weighted Resistor Network:

Fig. 17 shows the prime number weighted resistor network and an input current source applied to each node. Here

$R + R_{p2}$=2R

$R + R_{p2} + R_{p3}$=3R

$R + R_{p2} + R_{p3} + R_{p4}$=5R

$R + R_{p2} + R_{p3} + R_{p4} + R_{p5}$ =7R

$R + R_{p2} + R_{p3} + R_{p4} + R_{p5} + R_{p6}$=11R,

$R + R_{p2} + R_{p3} + R_{p4} + R_{p5} + R_{p6} + R_{p7}$=13R

$R + R_{p2} + R_{p3} + R_{p4} + \cdots + R_{p(K+1)}$

= [K-th prime number] R

Notice that 2, 3, 5, 7, 11, 13 are prime numbers. Then we have the following:

$R_{p2} = R, R_{p3} = R, R_{p4} = 2R, R_{p5} = 2R,$

$R_{p6} = 4R, R_{p7} = 3$

$R_{p(K+1)}$ = {K-th prime number –

(1+$\sum_{i=1}^{K-1}$[$(K-1)$-th prime number])} R

(K=1, 2, 3, …)

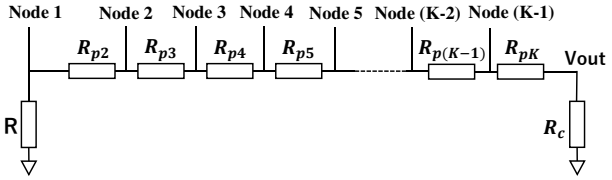


Fig. 17 Designed prime number weighted resistor network.

We have the following in Fig. 18 from Fig. 4 (a) and Eq. (1):

$R_a + R_b = R + R + R + 2R + 2R + 4R + 2R = 13R$

$V_{OUT} = I\,R_a\,[R_c\,/\,(R_a + R_b + R_c)]$.

Also $R_c = R$, and then $V_{OUT} = \left(\frac{1}{14}\right) R_a I$.

In Fig. 18 (a), $R_a = R$, and $V_{OUT1} = \left(\frac{1}{14}\right) RI$.

In Fig. 18 (b), $R_a = R + R = 2R$, and $V_{OUT2} = \left(\frac{2}{14}\right) RI$.

In Fig. 18 (c), $R_a = R + R + R = 3R$, and

$V_{OUT3} = \left(\frac{3}{14}\right) RI$.

In Fig. 18 (d), $R_a = R + R + R + 2R = 5R$,

and $V_{OUT5} = \left(\frac{5}{14}\right) RI$.

In Fig. 18 (e), $R_a = R + R + R + 2R + 2R = 7R$,

and $V_{OUT7} = \left(\frac{7}{14}\right) RI$.

In Fig. 18 (f), $R_a = R + R + R + 2R + 2R + 4R$

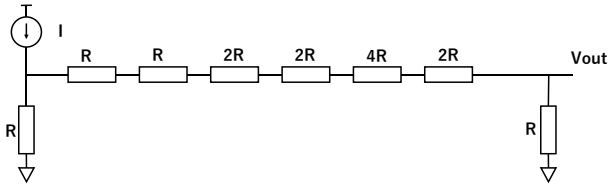
$= 11R$, and $V_{OUT11} = \left(\frac{11}{14}\right) RI$.

In Fig. 18 (g),

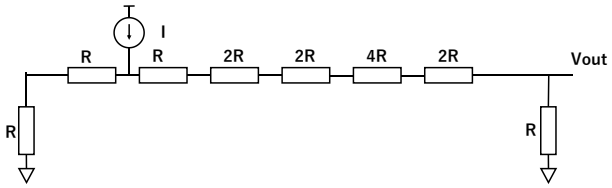
$R_a = R + R + R + 2R + 2R + 4R + 2R = 13R$,

and $V_{OUT13} = \left(\frac{13}{14}\right) RI$.

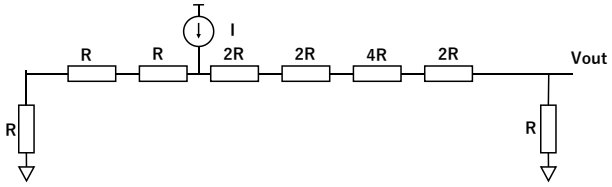
Fig. 19 (a) shows the case that the digital input is 6 while Fig. 19 (b) is the case it is 8. Note that 5+7 = 2x6 and 5+11 = 2x8. Fig. 20 shows the SPICE simulation verification.



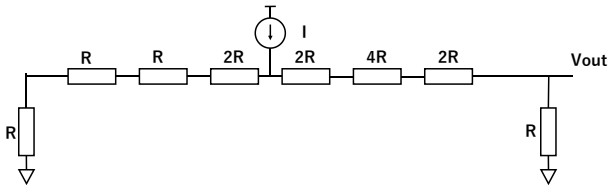
(a)



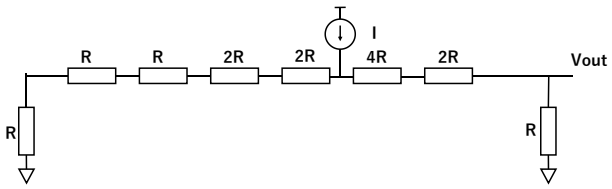
(b)



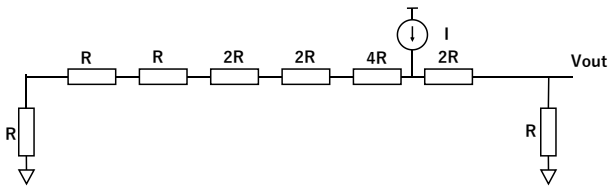
(c)



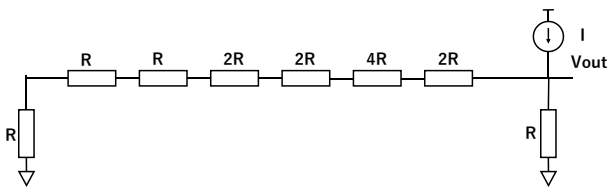
(d)



(e)

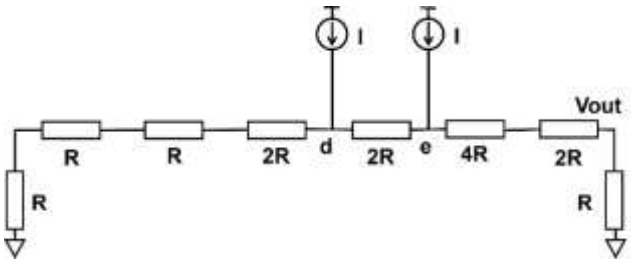


(f)

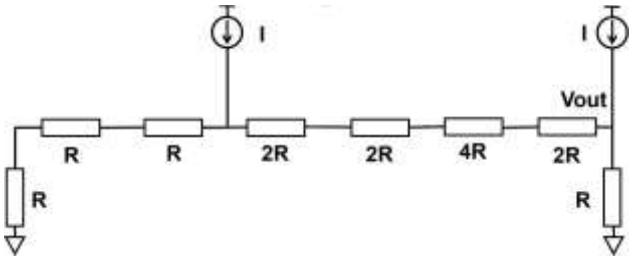


(g)

Fig. 18 Operation of the prime number weighted resistor network, which generates the prime number weighted voltage. (a) V_{OUT} = (1/14) RI. (b) V_{OUT} = (2/14) RI. (c) V_{OUT} = (3/14) RI. (d) V_{OUT} = (5/14) RI. (e) V_{OUT} = (7/14) RI. (f) V_{OUT} = (11/14) RI.. (g) V_{OUT} = (13/14) RI.

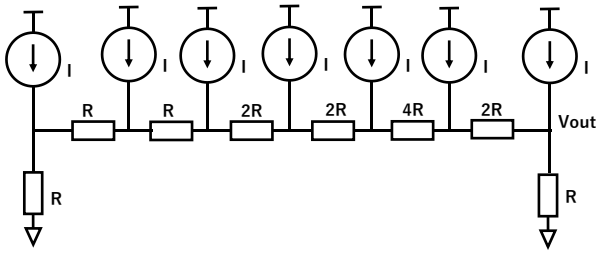


(a)

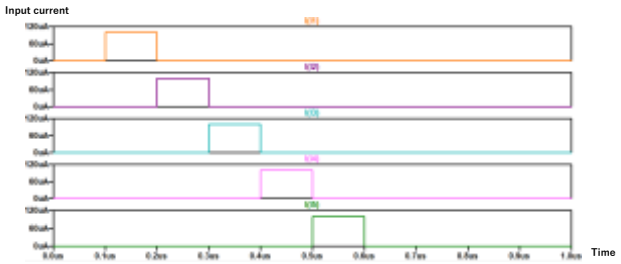


(b)

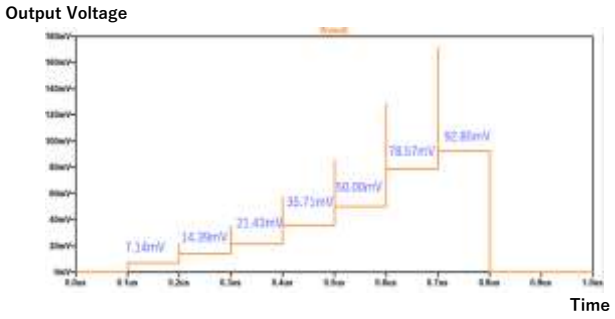
Fig. 19 Prime number DAC operation. (a) Digital input = 6 and Vout = (6/7) RI. (b) Digital input = 8 and Vout = (8/7) RI.



(a)



(b)



(c)

Fig. 20 SPICE simulation of the proposed prime number weighted resistor network. (a) Simulation circuit. (b) Input current source waveforms. (c) Output voltage (V_{OUT}) waveform.

Switch Arrays and Decoder Design:

We have also designed a decoder logic for a prime number DAC. Its digital inputs are from 0 to 63, and two switch arrays corresponding to two current sources is represented by Sa and Sb. There are 23 switches in the switch array of each current source, which are represented by Sa00... Sa22, Sb00... Sb22.

Due to the large number, we cannot verify the assumptions one by one, so we write an automatic verification program to detect whether all the input conditions are in line with our conjecture. The following is the design principle of calibration program:

Suppose that the digital input is 10 in decimal or B5, B4, B3, B2, B1 and B0 are 000101 in binary. Notice that 3+7=10 as shown in Fig. 19. Then three switches of Sa1 and Sb1 are ON (Sa3 =3, Sb5 = 7) and the other switches are OFF, as shown in Figs. 21, 22.

The prime number DAC decoder has the digital inputs of B5, B4, B3, B2, B1, B0 (from 0 to 63) and the digital outputs of Sa00, Sa01, Sa02, … Sa22, Sb00, …Sb22. Each logical expression is verified by a C program with the following algorithm:

Data A, Data B, and Data are defined by

$$\text{DataA} = 0 \cdot \text{Sa00} + 1 \cdot \text{Sa01} + 2 \cdot \text{Sa02} + 3 \cdot \text{Sa03} + 5 \cdot \text{Sa04} + 7 \cdot \text{Sa05} + 11 \cdot \text{Sa06} + \cdots$$

$$\text{DataB} = 0 \cdot \text{Sb00} + 1 \cdot \text{Sb01} + 2 \cdot \text{Sb02} + 3 \cdot \text{Sb03} + 5 \cdot \text{Sb04} + 7 \cdot \text{Sb05} + 11 \cdot \text{Sb06} + \cdots$$

$$\text{Data} = \text{B0} + 2 \cdot \text{B1} + 4 \cdot \text{B2} + 8 \cdot \text{B3} + 16 \cdot \text{B4} + 32 \cdot \text{B5}.$$

Here Sa00, Sa01, Sa02, … Sa22, Sb00, …Sb22, Sc00 are written as designed decoder logical expressions of B5, B4, B3, B2, B1, B0.

The program runs from B5 B4 B3 B2 B1 B0 = 000000 to 11111 and verified the following in all cases.

If $2 * \text{Data} = \text{DataA} + \text{DataB}$, the output is OK.

If $2 * \text{Data} != \text{DataA} + \text{DataB}$, the output is WRONG.

When for all B5, B4, B3, B2, B1, B0, the output is OK, then the decoder logical expressions are correct. The verification program execution results are as shown as Fig. 23.

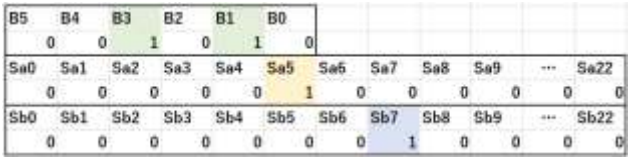


Fig. 21 DAC decoder logic for the digital input of 10.

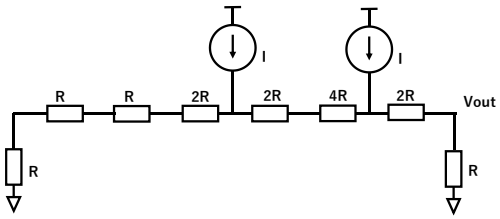


Fig. 22 Two current sources connection to the prime number weighted network for digital input of 10.

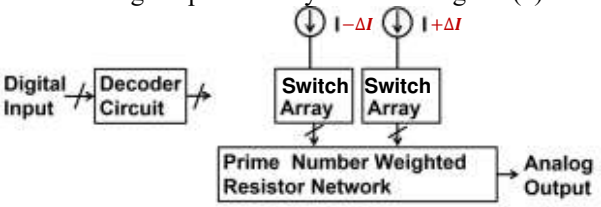
```
B5=0, B4=0, B3=0, B2=0, B1=0, B0=0   OK   0=0+0
B5=0, B4=0, B3=0, B2=0, B1=0, B0=1   OK   2=2+0
B5=0, B4=0, B3=0, B2=0, B1=1, B0=0   OK   4=1+3
B5=0, B4=0, B3=0, B2=0, B1=1, B0=1   OK   6=3+3
B5=0, B4=0, B3=0, B2=1, B1=0, B0=0   OK   8=3+5
B5=0, B4=0, B3=0, B2=1, B1=0, B0=1   OK   10=3+7
B5=0, B4=0, B3=0, B2=1, B1=1, B0=0   OK   12=5+7
B5=0, B4=0, B3=0, B2=1, B1=1, B0=1   OK   14=7+7
B5=0, B4=0, B3=1, B2=0, B1=0, B0=0   OK   16=5+11
B5=0, B4=0, B3=1, B2=0, B1=0, B0=1   OK   18=7+11
B5=0, B4=0, B3=1, B2=0, B1=1, B0=0   OK   20=7+13
```

Fig. 23 Execution result of the prime number DAC decoder verification program.

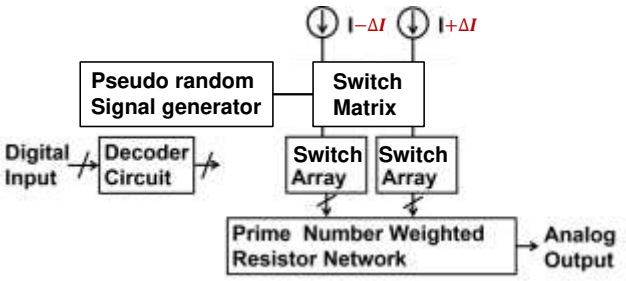
Remark 2: According to the Goldbach conjecture, any even number is the sum of two prime numbers. However, notice that its expression is not unique; a given even number n can be represented by $n = n_1 + n_2$ and $n = m_1 + m_2$ where n_1, n_2, m_1 and m_2 are prime numbers. For example, in case n=18, 18=5+13 and 18=7+11. For the decoder design, we choose one of them for a given input data.

Dynamic Element Matching (DEM) Techniques [3, 12]

(i)The prime number DAC has two current sources and in actual circuit, they can have mismatches ΔI as shown in Fig. 24 (a). However, using the pseudo-random switching, the mismatches can be time-averaged and the spurious components due to them can be spread out in frequency domain using simple circuitry as shown Fig. 24 (b).



(a)



(b)

Fig. 24 Current source mismatches and the prime number DAC with dynamic element matching circuit. (a) Current source mismatches. (b) Dynamic element matching.

(ii) Also the resistors can have mismatches as shown in Fig. 25. However, as the above "Remark 2" says, for a given input data, there can be multiple expressions of two-prime number sum. Suppose that the input is DC and its value is 9, and at time n, 9x2=5+13 is used while at time n+1, 9x2=7+11 is used. If such selections are done dynamically in a pseudo-random manner with modified decoder design, the resistor mismatch effects may be time-averaged and also the spurious components due to them may be spread out in frequency domain.

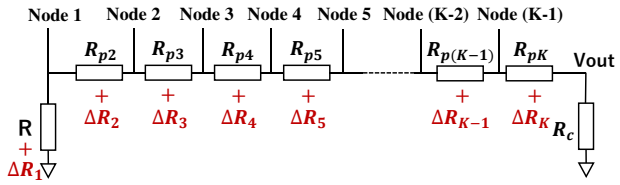


Fig. 25 Resistor mismatches in the prime number DAC.

Application of these DEM techniques to the prime number DACs and also the N-polygonal number DACs should be further investigated as the next step.

4. Conclusion

This paper has demonstrated that new DAC architectures can be derived, based on number theory by integrating the knowledge of mathematics and physical electricity, and their operations are verified by simulation; the theoretical conjecture is consistent with the simulation results. Integers have very interesting properties, but they have not been fully exploited yet for the mixed-signal system and circuit design. We conclude this paper by remarking that in most cases, mixed-signal architecture design is based on designer's experiences but not mathematics, and the attempt of its new architecture derivation from mathematics may have possibilities to lead to very new ones. As the next step, we will consider the derivation of the DAC architectures considering their practical aspects.

Acknowledgments and Funding Declaration

This work was supported by JSPS KAKENHI Grant Number 21K04190.

Data Availability

Data sharing will be made available with reasonable request.

References

[1] G. Manganaro, Advanced Data Converters, Cambridge University Press (2011).
[2] F. Maloberti, Data Converters, Springer (2007).
[3] R. V. D. Plassche, CMOS Integrated Analog-to-Digital and Digital-to-Analog Converters, Springer (2003).
[4] The Data Conversion Handbook, Analog Devices Inc. (2005).
[5] H. G. Dimopoulos, Analog Electronic Filters: Theory, Design and Synthesis, Springer (2011).
[6] M. Leonard Eugene Dickson, "History of the Theory of Numbers," Diophantine Analysis, Dover, Vol. 2 (2005).
[7] H. Kobayashi, et. al., "Analog/Mixed-Signal Circuit Design in Nano CMOS Era", IEICE Electronic Express (2014).
[8] H. Kobayashi, et. al., "Analog/Mixed-Signal Circuit Testing Technologies in IoT Era", IEEE ICSICT (Nov. 2020).
[9] H. Kobayashi, et. al., "Analog/Mixed-Signal/RF Circuits for Complex Signal Processing", IEEE ASICON (Oct. 2019).
[10] H. Kobayashi, et. al., "Fine Time Resolution TDC Architectures-Integral and Delta-Sigma Types", IEEE ASICON (Oct. 2019).
[11] H. Kobayashi, H. Lin, "Analog/Mixed-Signal Circuit Design Based on Mathematics", IEEE ICSCIT (Oct. 2016).
[12] H. Kobayashi, et. al., "Performance Improvement of Delta-Sigma ADC/DAC/TDC Using Digital Technique", IEEE ICSICT (Nov. 2018).
[13] H. Kobayashi, et. al., "Unified Methodology of Analog/Mixed-Signal IC Design Based on Number Theory", IEEE ICSICT (Nov. 2018).
[14] Y. Kobayashi, H. Kobayashi, "Redundant SAR ADC Algorithm Based on Fibonacci Sequence", Advanced Micro-Device Engineering VI, Key Engineering Materials, pp.117-126 (2016).
[15] S. Yamamoto, et. al., "Metallic Ratio Equivalent-Time Sampling and Application to TDC Linearity Calibration", IEEE Trans. Device and Materials Reliability, Early Access (March 2022).
[16] S. Yamamoto, et. al., "Metallic Ratio Equivalent-Time Sampling: A Highly Efficient Waveform Acquisition Method", IEEE IOLTS (June 2021).
[17] Y. Sasaki, et. al., "Highly Efficient Waveform Acquisition Condition in Equivalent-Time Sampling System", IEEE Asian Test Symposium (Oct. 2018).

[18] S. Katayama, et. al., "Application of Residue Sampling to RF/AMS Device Testing", 30th IEEE Asian Test Symposium (Nov. 2021).

[19] Y. Abe, et. al., "Frequency Estimation Sampling Circuit Using Analog Hilbert Filter and Residue Number System", 13th IEEE International Conference on ASIC (Oct. 2019).

[20] Y. Zhao, et. al., "Revisit to Histogram Method for ADC Linearity Test: Examination of Input Signal and Ratio of Input and Sampling Frequencies", Journal of Electronic Testing: Theory and Applications, Springer, on-line published (March 2022).

[21] M. Hirai, et. al., "Digital-to-Analog Converter Configuration Based on Non-uniform Current Division Resistive-Ladder", ITC-CSCC (June 2021).

[22] M. Hirai, et. al., "Nonlinearity Analysis of Resistive Ladder-Based Current-Steering Digital-to-Analog Converter", IEEE ISOCC (Oct. 2020).

[23] M. Hirai, et. al., "Systematic Construction of Resistor Ladder Network for N-ary DACs", IEEE ASICON (Oct. 2019).

[24] R. Jiang, et. al., "Gray-code Input DAC Architecture for Clean Signal Generation", IEEE ISPACS (Nov. 2017).

[25] G. Adhikari, et. al., "Study of Gray Code Input DAC Using MOSFETs for Glitch Reduction", IEEE ICSICT (Oct. 2016).

[26] D. Yao, et. al., "Digital-to-Analog Converter Linearity Improvement Technique Based on Classical Number Theory for Modern ULSI", International Workshop on Post-Binary ULSI Systems (May 2021).

[27] M. Higashino, et. al., "Digital-to-Analog Converter Layout Technique and Unit Cell Sorting Algorithm for Linearity Improvement Based on Magic Square", Journal of Technology and Social Science (Jan. 2020).

[28] D. Yao, et. al., "DAC Linearity Improvement with Layout Technique using Magic and Latin Squares", IEEE ISPACS (Nov. 2017).

[29] D. Yao, et. al., "Segmented DAC Unit Cell Selection Algorithm and Layout/Routing Based on Euler's Knight Tour", International Conference on Analog VLSI Circuits (Oct. 2021).

[30] Y. Du, et. al., "Digital-to-Analog Converter Architectures Based on Polygonal and Prime Numbers", IEEE ISOCC (Oct. 2020).

[31] X. Bai, et. al., "Digital-to-Analog Converter Architectures Based on Goldbach Conjecture for Prime Numbers in Mixed-Signal ULSI", International Workshop on Post-Binary ULSI Systems (May 2021).

[32] X. Bai, et. al., "Design of Digital-to-Analog Converter Architectures Based on Polygonal Numbers", International Conference on Analog VLSI Circuits (Oct. 2021).