International Conference on Analog VLSI Circuits (AVIC)
Bordeaux, France
19th Oct. 2021

A Proposal of Digital-to-Analog Converter Architectures Based on Polygonal Numbers

Xueyan Bai, D. Yao, Y. Du, T. Tran,
A. Kuwana, H. Kobayashi,
K. Kubo
Gunma University
Oyama National College of Technology

Kobayashi Laboratory

群馬大学
GUNMA UNIVERSITY

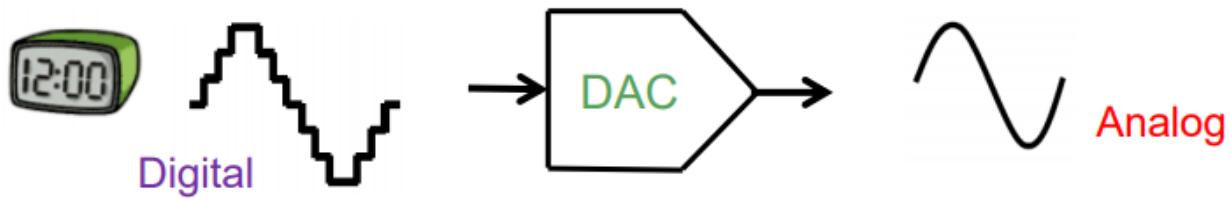
Contents

Contents

Research Objective

- **Interesting properties of number theory**

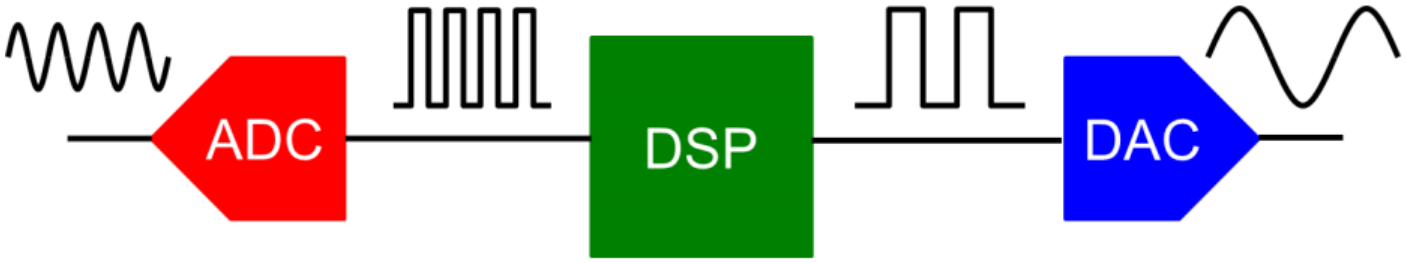
- **Possibility of new configurations of DAC（6-bit case）**



Contents

Importance of ADC / DAC

- **Rapid development of digital electronics technology**
- **A natural signal is analog**



DACs are Everywhere !

Communication equipment

Electronic measuring instrument

Audio systems

Integer Theory and Electronic Circuit Design

Many interesting properties of Integers

Currently
No Link

Electronic circuit designs

Our research here makes their links !

Carolus Fridericus Gauss
(1777-1855)

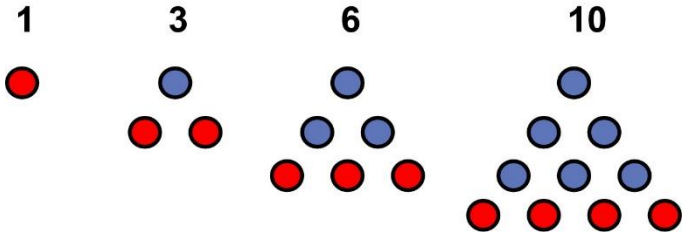
Integer theory is queen of Mathematics

Contents

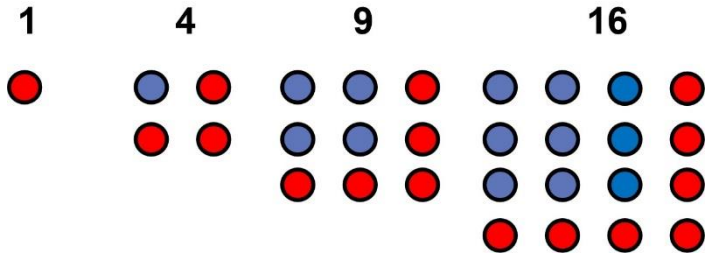
Polygonal Number

Polygonal Number

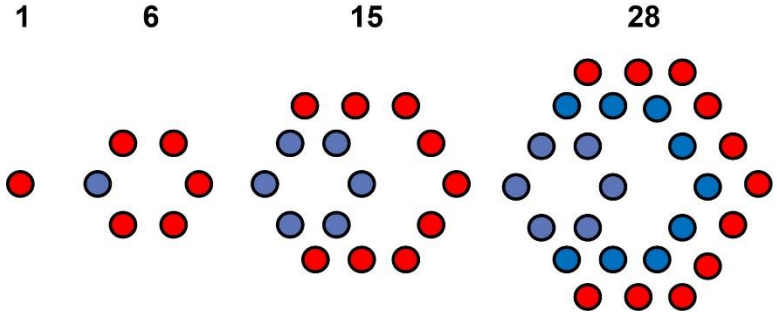
Represented as dots or pebbles arranged in the shape of a regular polygon.



(a)Triangular numbers.



(b) Square numbers.



(c) Hexagonal numbers.

Fermat Polygonal Number Theorem

Any natural number

↓ expressed by

Sum of N N-angular numbers

Pierre de Fermat
1607 – 1665

k-th of N-angular number, m(N, k) can be expressed by

$$m(N, k) = (1/2)\, k\, [(N-2)k - (N-4)]$$

Then N-angular numbers are given by

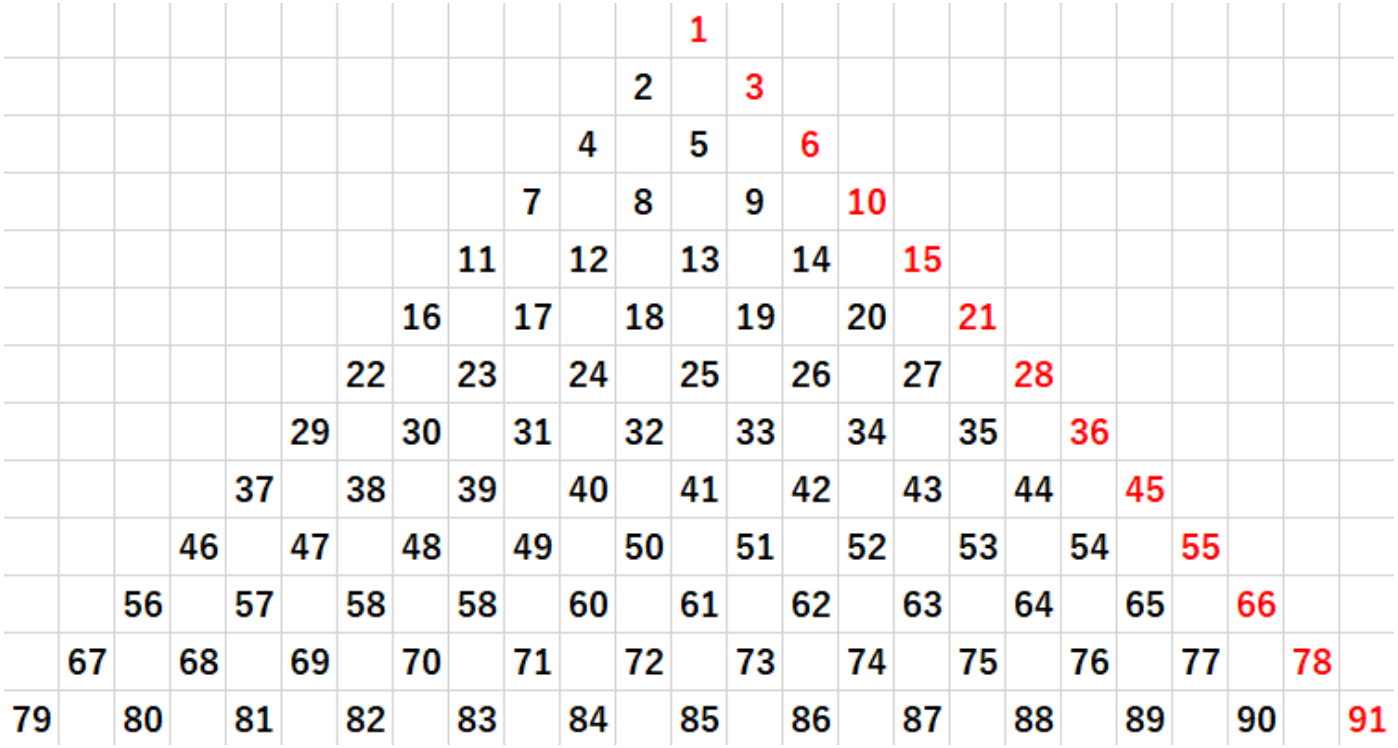
1, N, 3N-3, 6N-8, 10N-15, …

for k=1, 2, 3, 4, 5, …

Contents

What is Triangular Number ?

Triangular Number : 1, 3, 6, 10, 15, 21, 28, 36, 45, 55, 66, ..n(n+1)/2



Theory of Trigonometric Numbers

Any natural number → Sum of 3 triangular numbers

expressed by

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

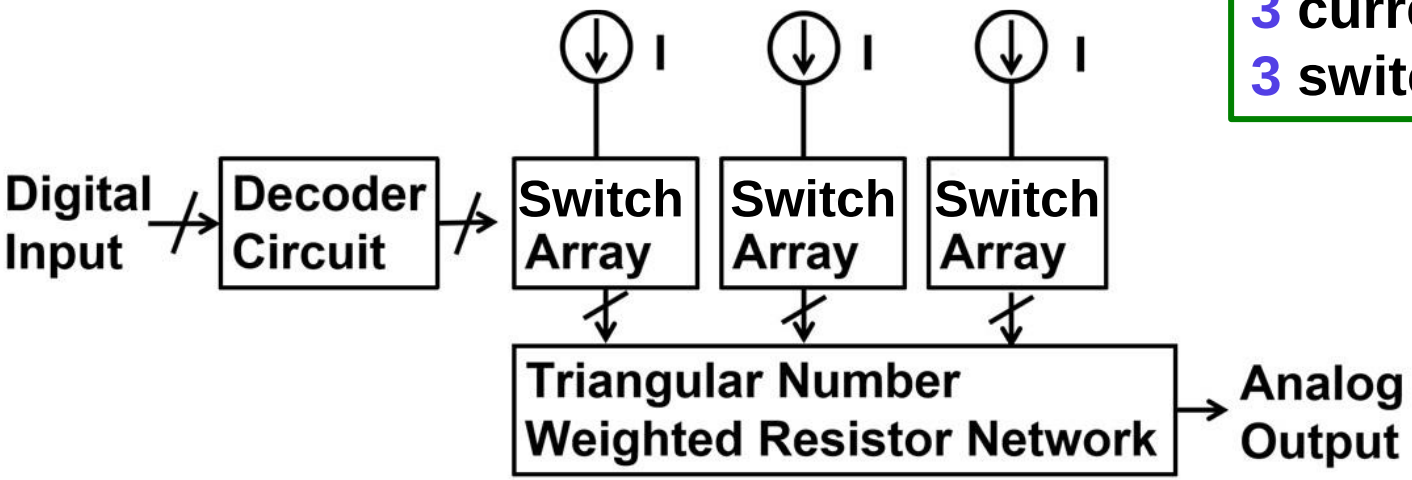
Theory of Trigonometric Numbers

Any natural number → Sum of 3 triangular numbers

expressed by

61:	6+55	76:	10+66	91:	91	106:	1+105	121:	1+120
62:	1+6+55	77:	1+10+66	92:	1+91	107:	1+1+105	122:	1+1+120
63:	3+15+45	78:	78	93:	1+1+91	108:	3+105	123:	3+120
64:	3+6+55	79:	1+78	94:	3+91	109:	1+3+105	124:	1+3+120
65:	10+55	80:	1+1+78	95:	1+3+91	110:	55+55	125:	10+10+105
66:	66	81:	3+78	96:	3+15+78	111:	6+105	126:	6+120
67:	1+66	82:	1+3+78	97:	6+91	112:	1+6+105	127:	1+6+120
68:	1+1+66	83:	28+55	98:	1+6+91	113:	1+21+91		
69:	3+66	84:	6+78	99:	21+78	114:	3+6+105		
70:	1+3+66	85:	1+6+78	100:	3+6+91	115:	10+105		
71:	6+10+55	86:	10+10+66	101:	10+91	116:	1+10+105		
72:	6+66	87:	3+6+78	102:	1+10+91	117:	6+6+105		
73:	1+6+66	88:	10+78	103:	6+6+91	118:	3+10+105		
74:	1+28+45	89:	1+10+78	104:	3+10+91	119:	28+91		
75:	3+6+66	90:	6+6+78	105:	105	120:	120		

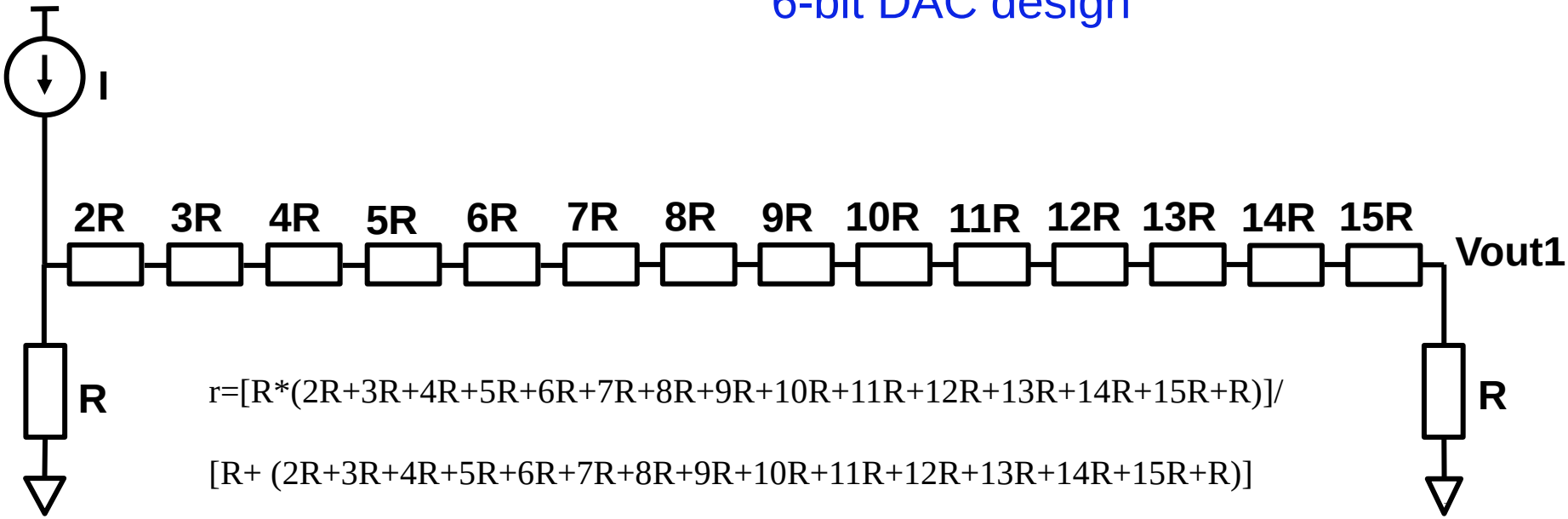
Proposed Triangular Number DAC



3 current sources
3 switch arrays

Triangular Number DAC Input 1

6-bit DAC design



r=[R*(2R+3R+4R+5R+6R+7R+8R+9R+10R+11R+12R+13R+14R+15R+R)]/

[R+ (2R+3R+4R+5R+6R+7R+8R+9R+10R+11R+12R+13R+14R+15R+R)]

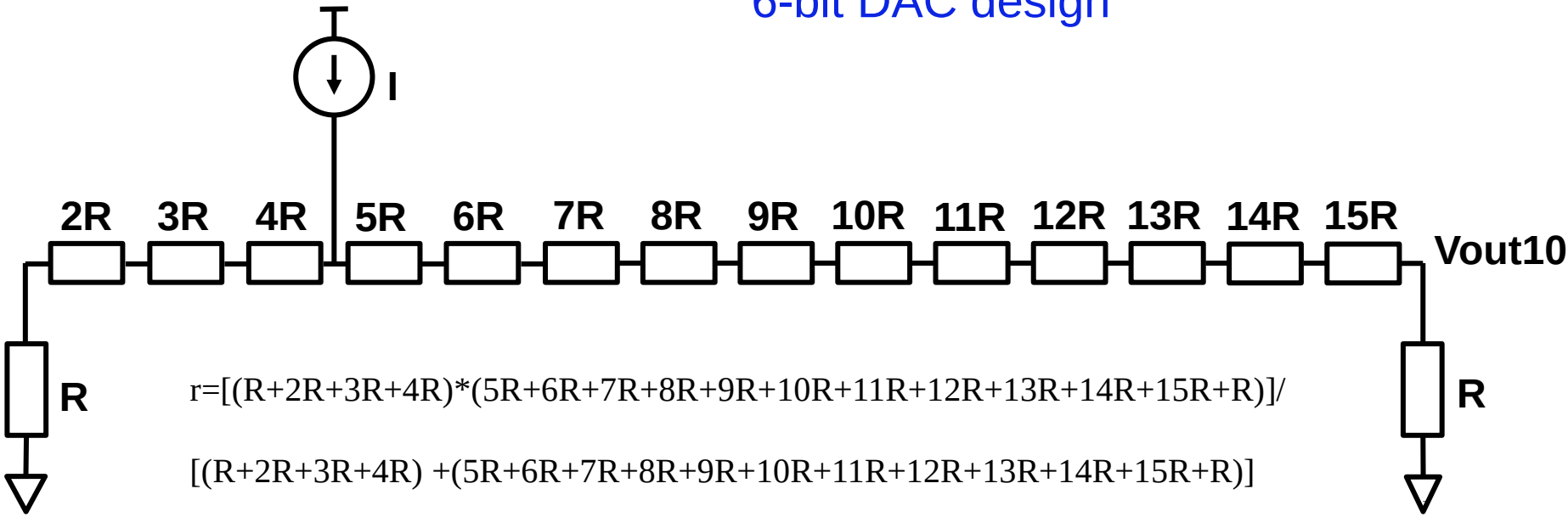
V=I*r

Vout1/V=R/(2R+3R+4R+5R+6R+7R+8R+9R+10R+11R+12R+13R+14R+15R+R)

Vout1=I*R/121

Triangular Number DAC Input 10

6-bit DAC design



r=[(R+2R+3R+4R)*(5R+6R+7R+8R+9R+10R+11R+12R+13R+14R+15R+R)]/

[(R+2R+3R+4R) +(5R+6R+7R+8R+9R+10R+11R+12R+13R+14R+15R+R)]

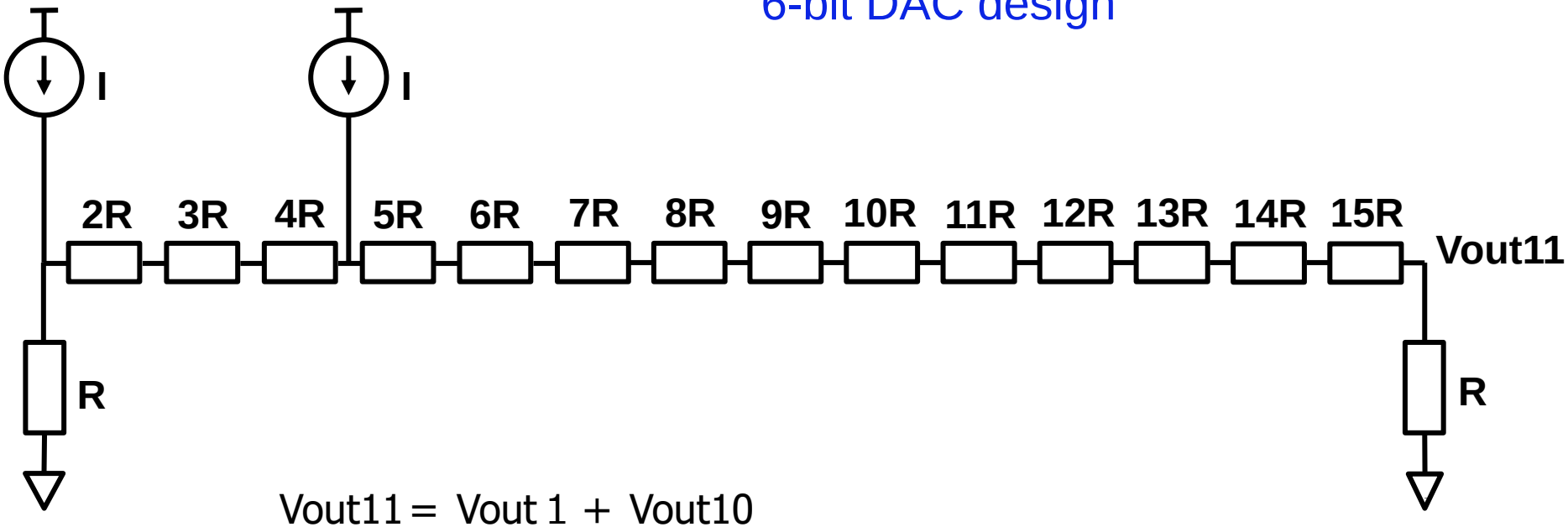
V=I*r

Vout10/V=R/(5R+6R+7R+8R+9R+10R+11R+12R+13R+14R+15R+R)

Vout10=10I*R/121

Triangular Number DAC Input 11

6-bit DAC design



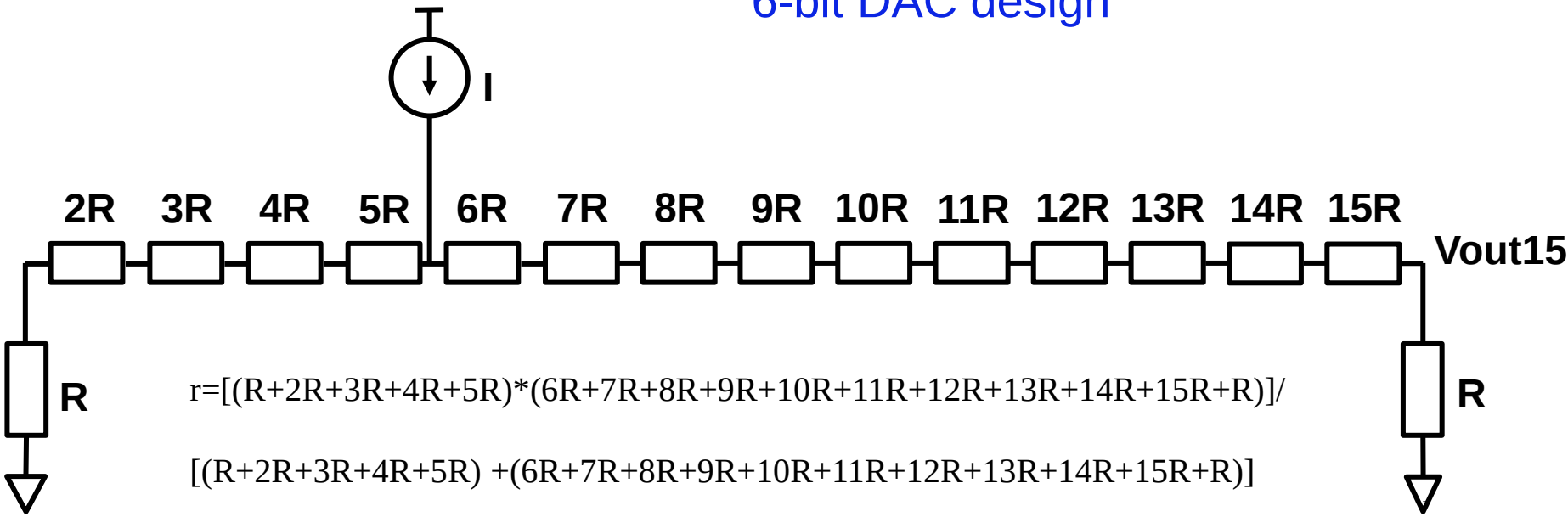
$$Vout11 = Vout1 + Vout10$$

$$= I*R/121 + 10I*R/121$$

$$= 11I*R/121$$

Triangular Number DAC Input 15

6-bit DAC design



r=[(R+2R+3R+4R+5R)*(6R+7R+8R+9R+10R+11R+12R+13R+14R+15R+R)]/

[(R+2R+3R+4R+5R) +(6R+7R+8R+9R+10R+11R+12R+13R+14R+15R+R)]

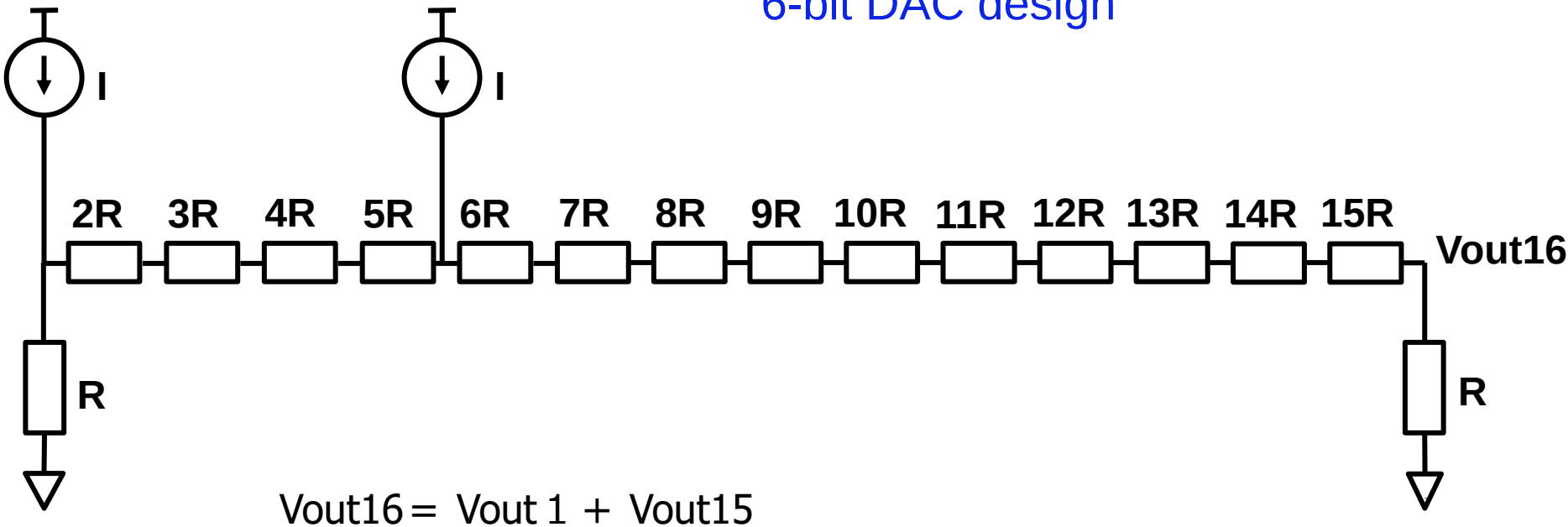
V=I*r

Vout15/V=R/(6R+7R+8R+9R+10R+11R+12R+13R+14R+15R+R)

Vout15=15I*R/121

Triangular Number DAC Input 16

6-bit DAC design



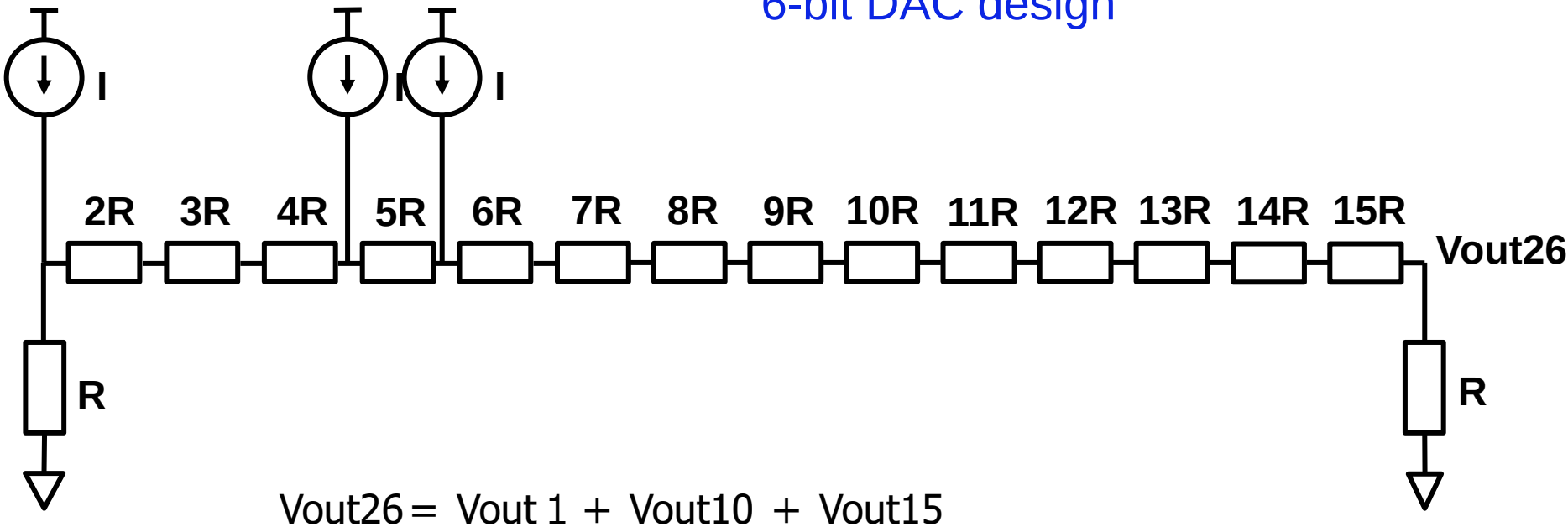
$$Vout16 = Vout\,1 + Vout15$$

$$= I*R/121 + 15I*R/121$$

$$= 16I*R/121$$

Triangular Number DAC Input 26

6-bit DAC design



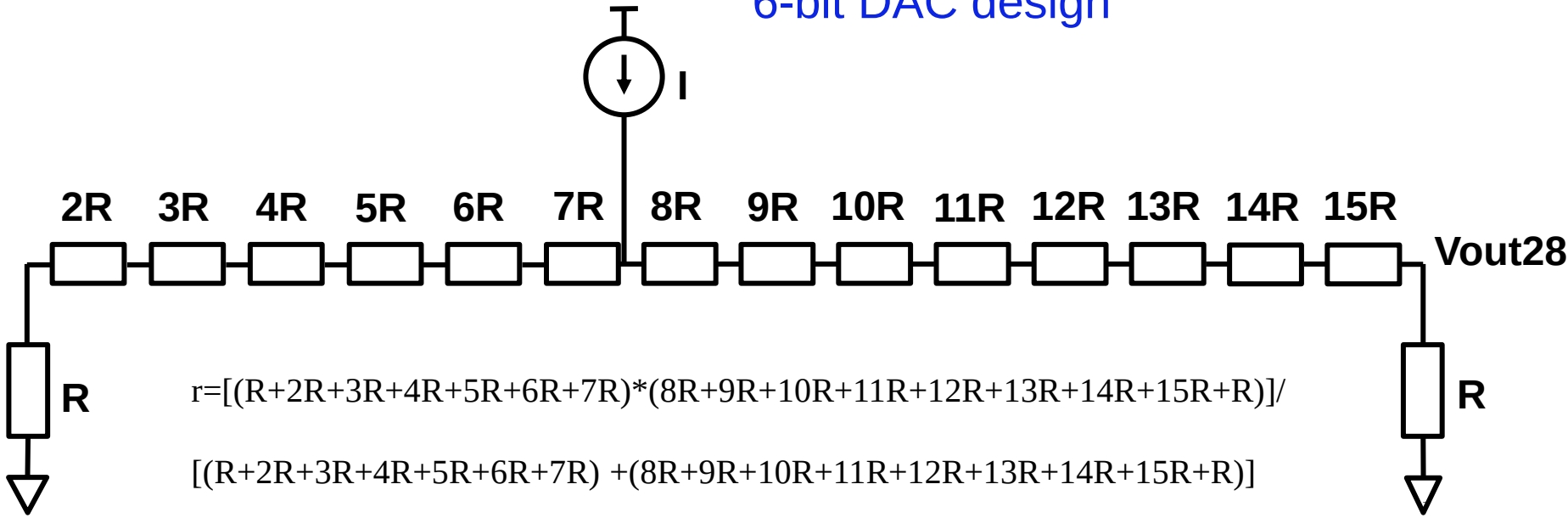
$$Vout26 = Vout1 + Vout10 + Vout15$$

$$= I*R/121 + 10I*R/121 + 15I*R/121$$

$$= 26I*R/121$$

Triangular Number DAC Input 28

6-bit DAC design



r=[(R+2R+3R+4R+5R+6R+7R)*(8R+9R+10R+11R+12R+13R+14R+15R+R)]/

[(R+2R+3R+4R+5R+6R+7R) +(8R+9R+10R+11R+12R+13R+14R+15R+R)]

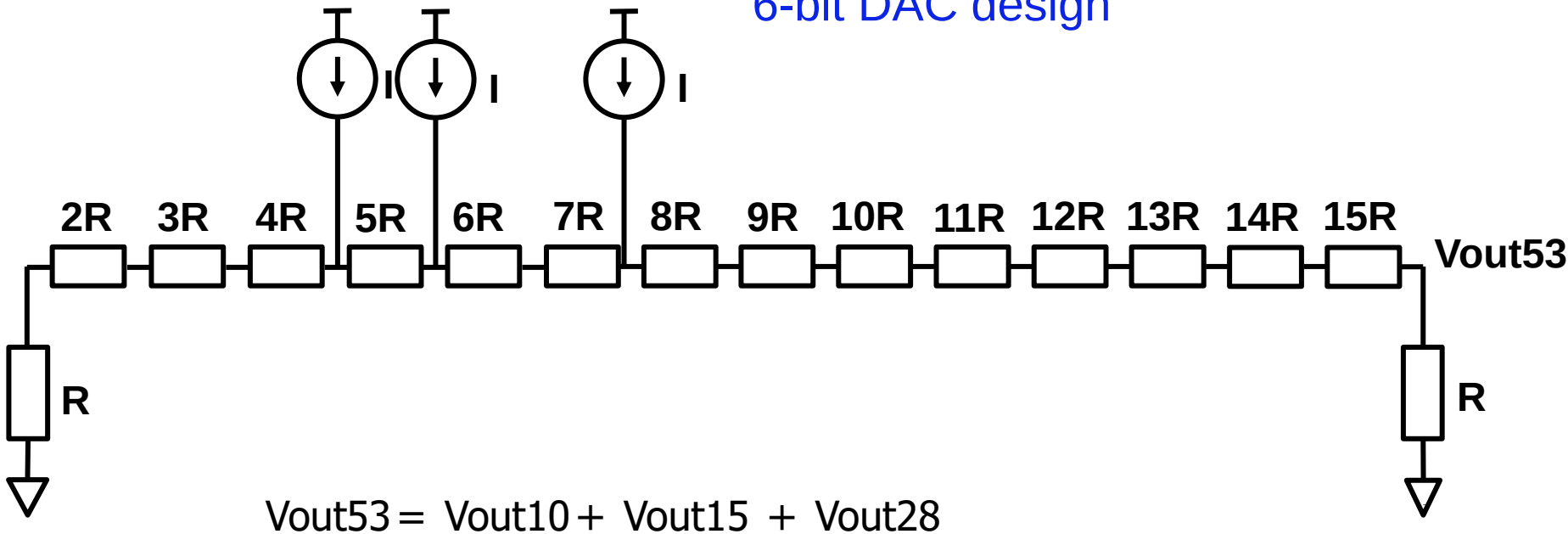
V=I*r

Vout28/V=R/(8R+9R+10R+11R+12R+13R+14R+15R+R)

Vout28=28I*R/121

Triangular Number DAC Input 53

6-bit DAC design



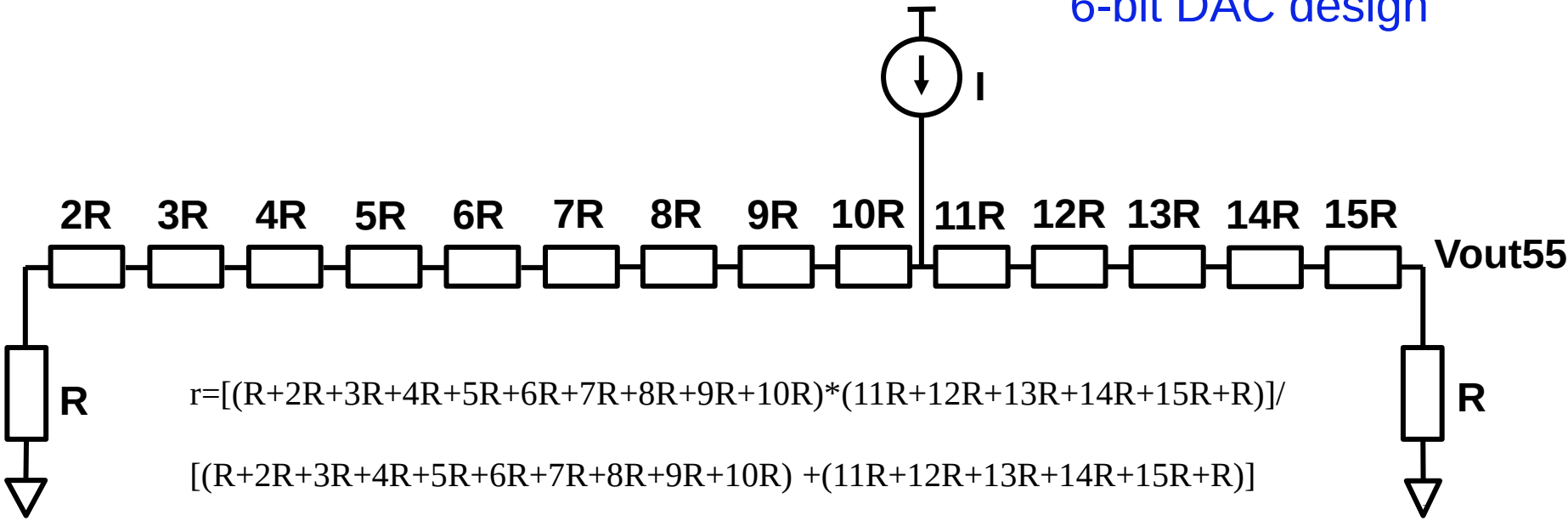
Vout53 = Vout10 + Vout15 + Vout28

= 10I*R/121 + 15I*R/121 + 28I*R/121

= 53I*R/121

Triangular Number DAC Input 55

6-bit DAC design



r=[(R+2R+3R+4R+5R+6R+7R+8R+9R+10R)*(11R+12R+13R+14R+15R+R)]/

[(R+2R+3R+4R+5R+6R+7R+8R+9R+10R) +(11R+12R+13R+14R+15R+R)]

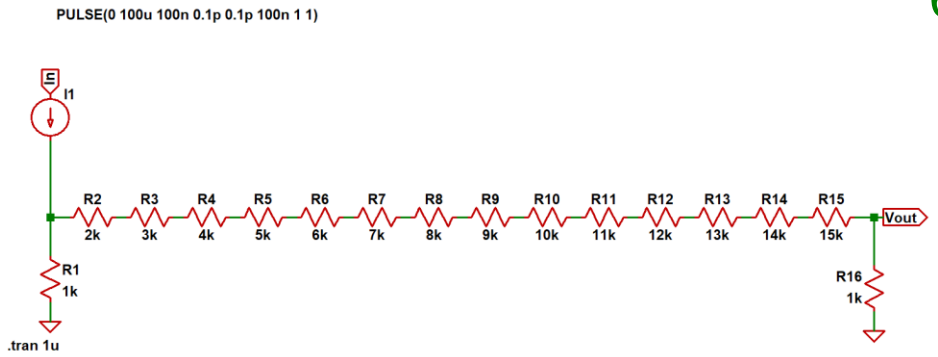
V=I*r

Vout55/V=R/(11R+12R+13R+14R+15R+R)

Vout55=55I*R/121

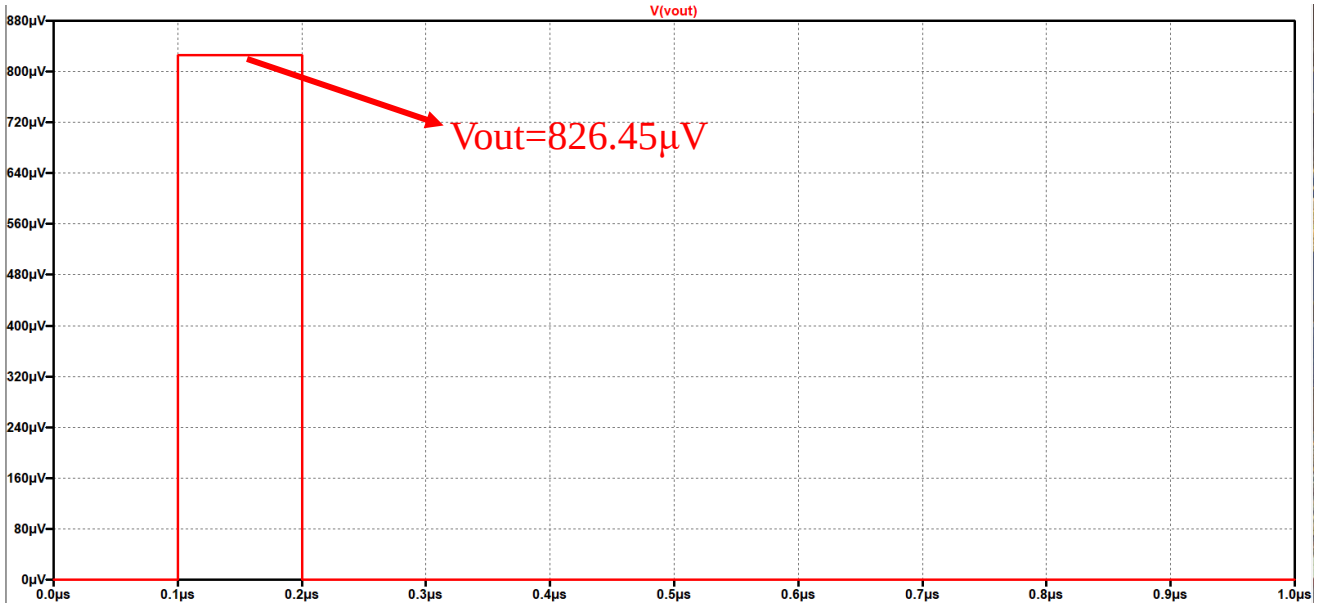
Triangular Number DAC Input 1

6-bit DAC simulation



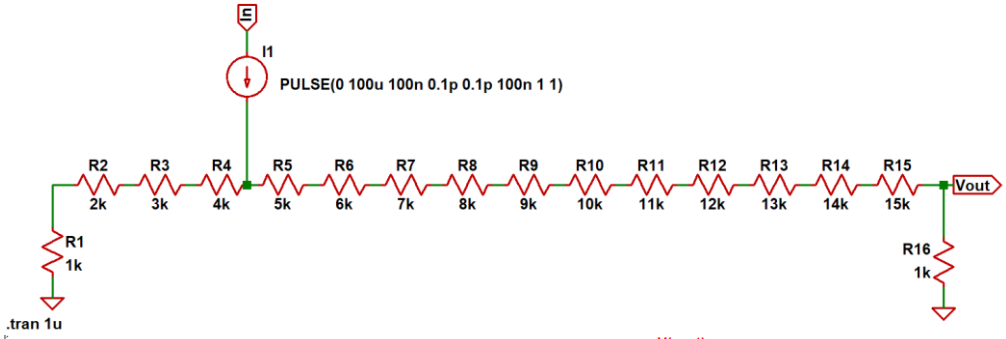
Theoretical Value Vout=I*R/121=826.45μV

Simulation Value Vout=826.45μV



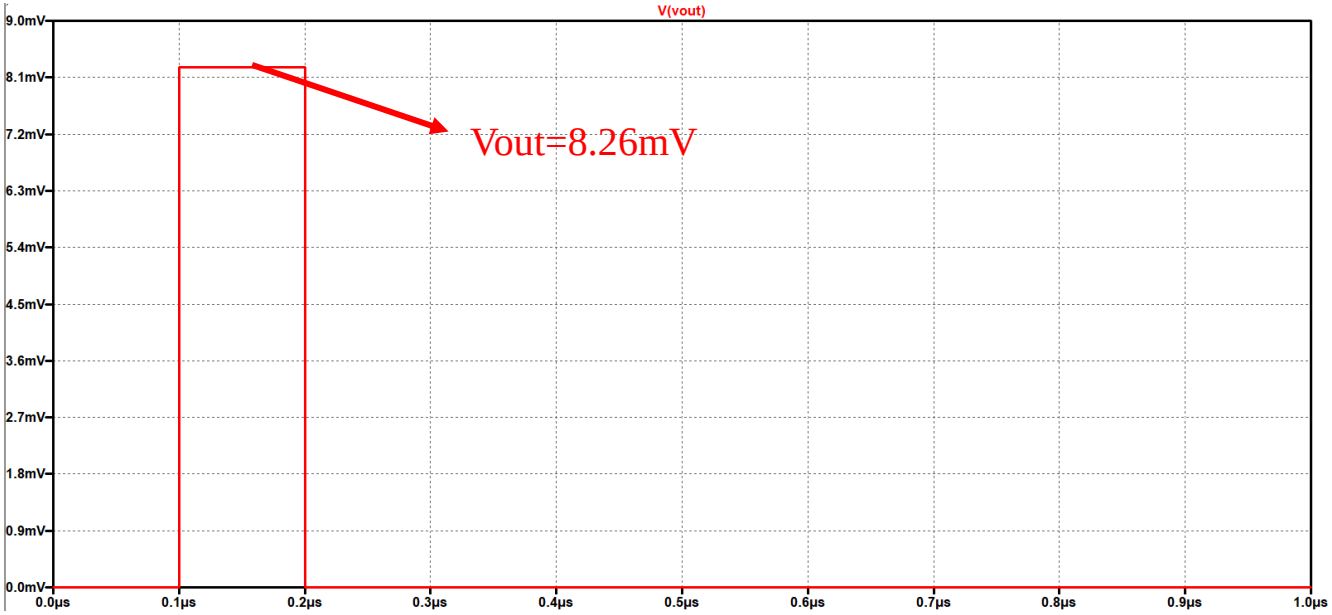
Triangular Number DAC Input 10

6-bit DAC simulation



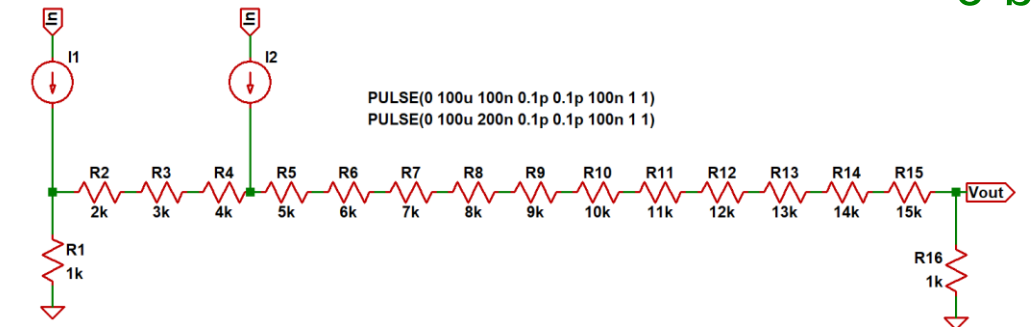
Theoretical Value Vout=10I*R/121=8.26mV

Simulation Value Vout=8.26mV



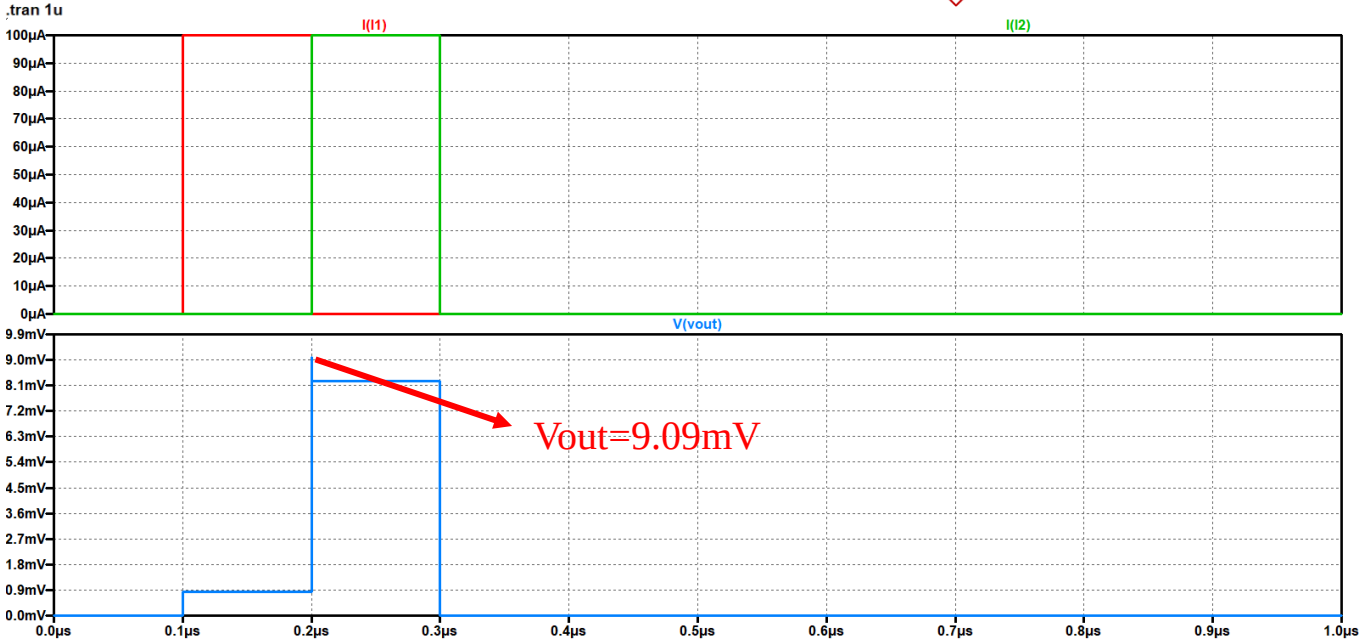
Triangular Number DAC Input 11

6-bit DAC simulation



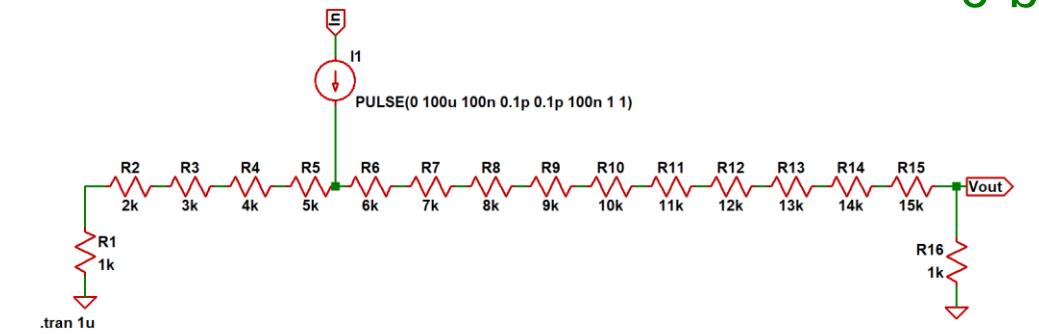
Theoretical Value Vout=11I*R/121=9.09mV

Simulation Value Vout=9.09mV



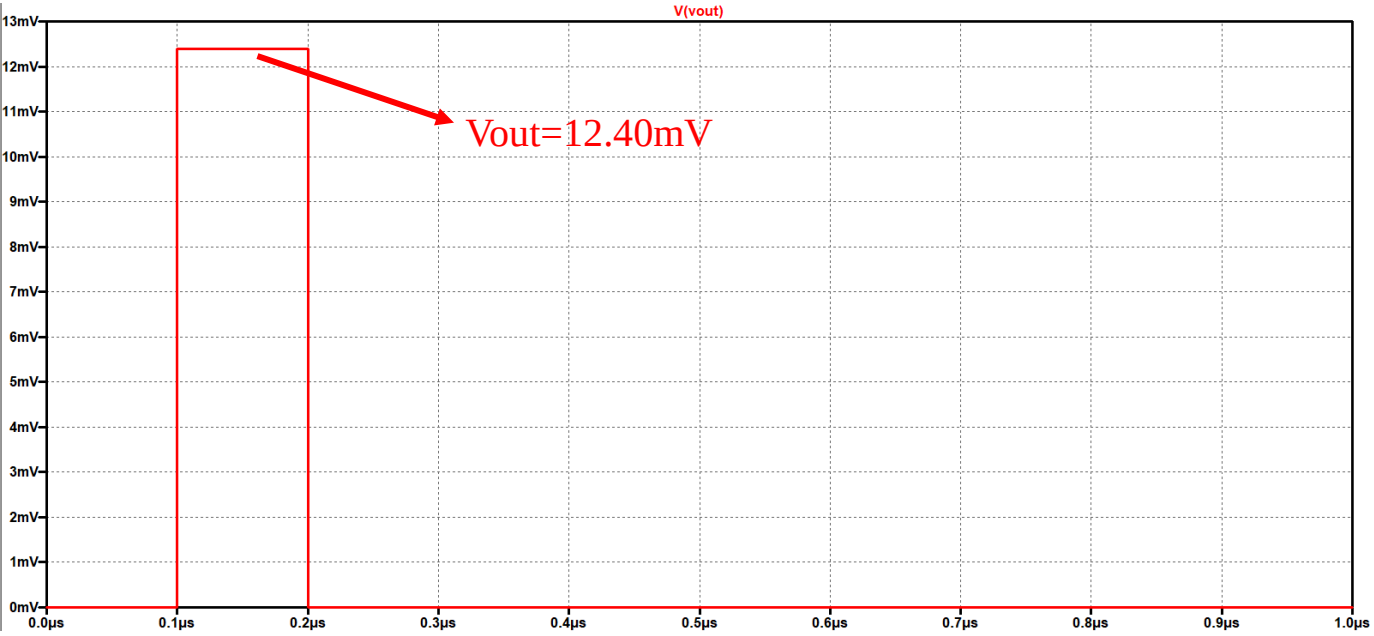
Triangular Number DAC Input 15

6-bit DAC simulation



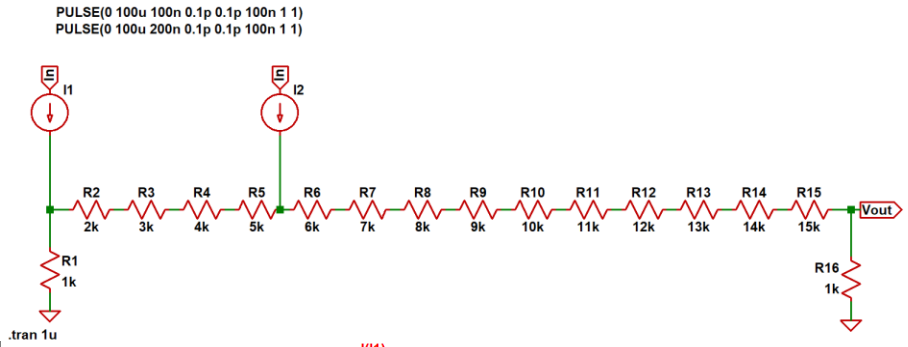
Theoretical Value Vout=15I*R/121=12.40mV

Simulation Value Vout=12.40mV



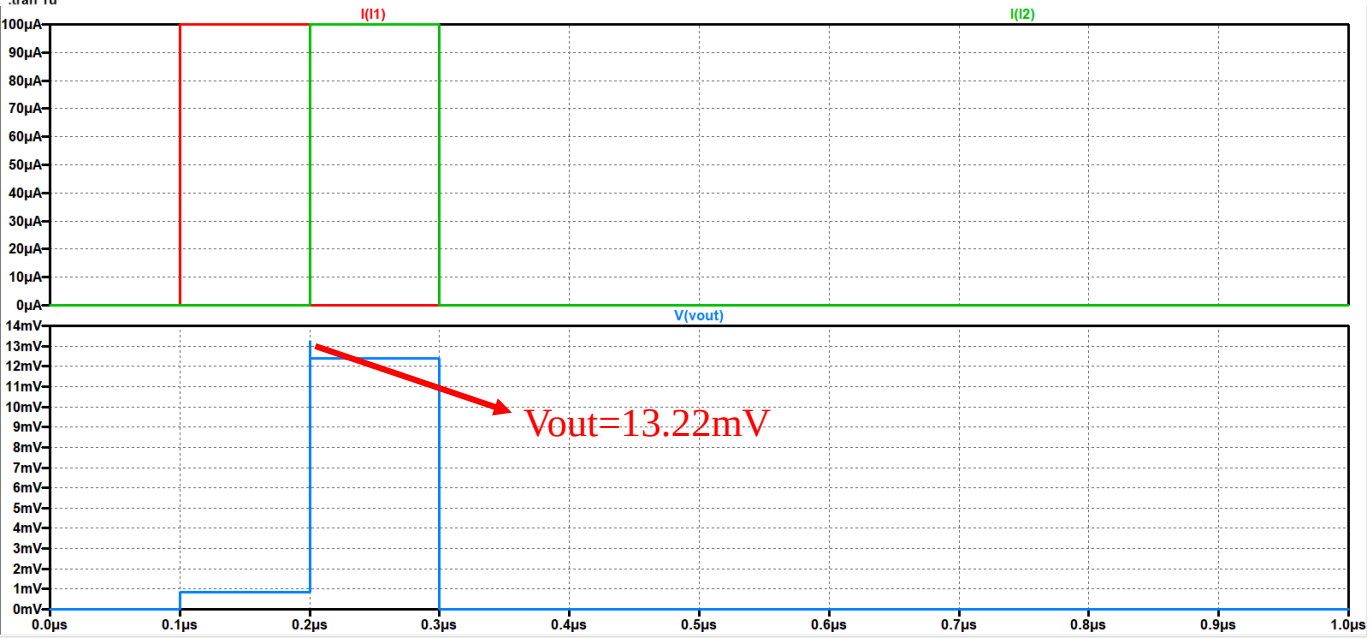
Triangular Number DAC Input 16

6-bit DAC simulation



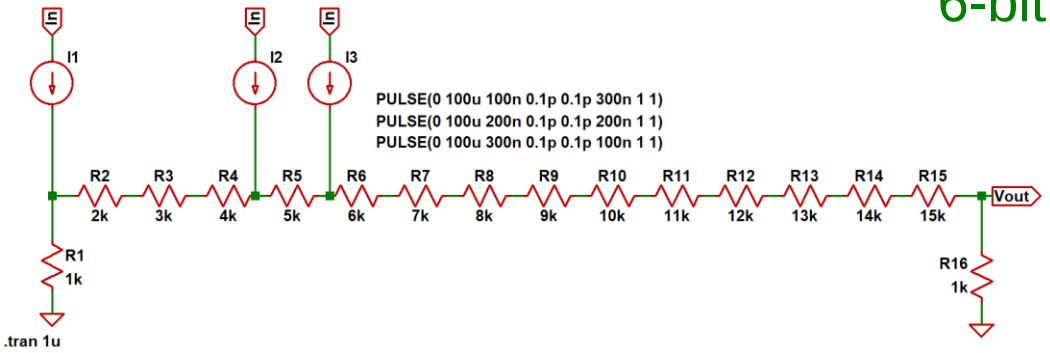
Theoretical Value Vout=16I*R/121=13.22mV

Simulation Value Vout=13.22mV



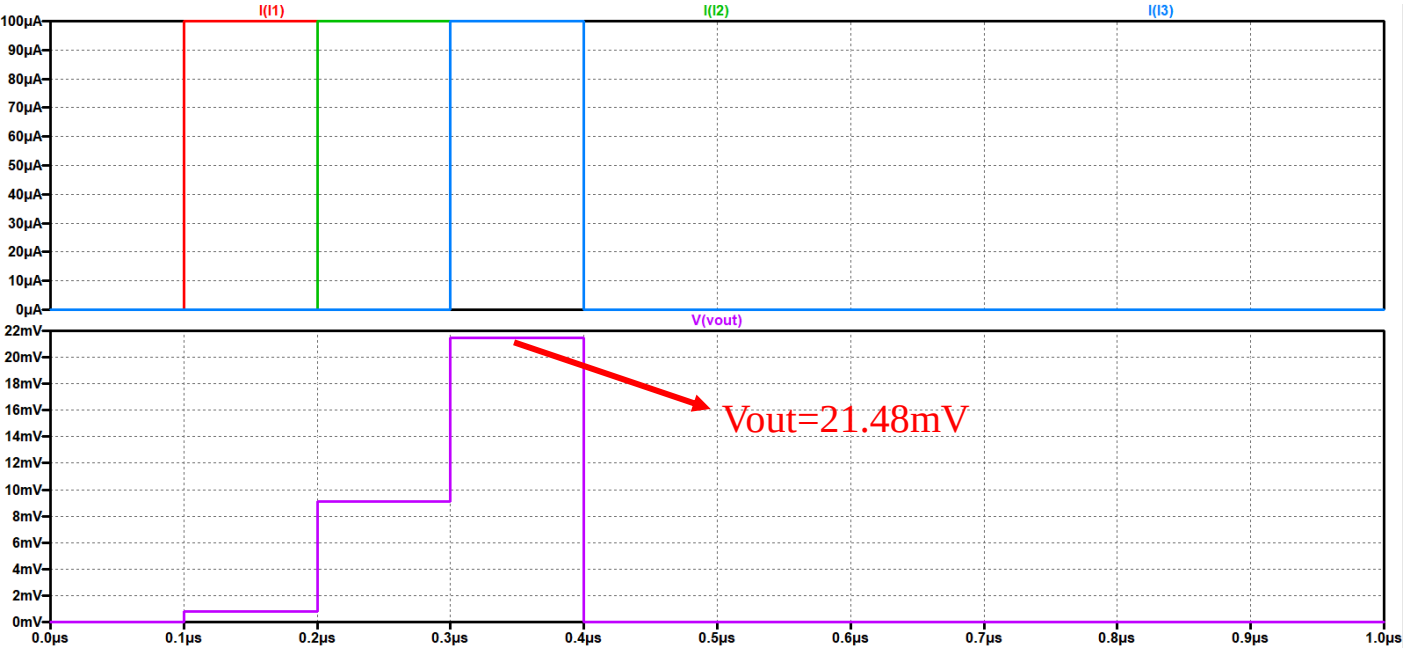
Triangular Number DAC Input 26

6-bit DAC simulation



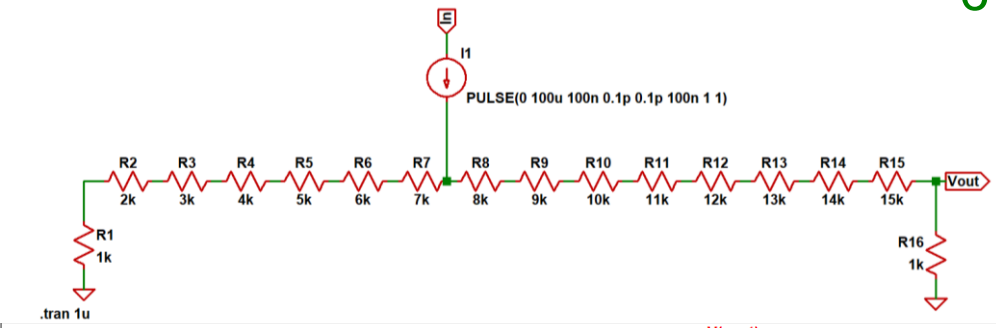
Theoretical Value Vout=26I*R/121=21.48mV

Simulation Value Vout=21.48mV



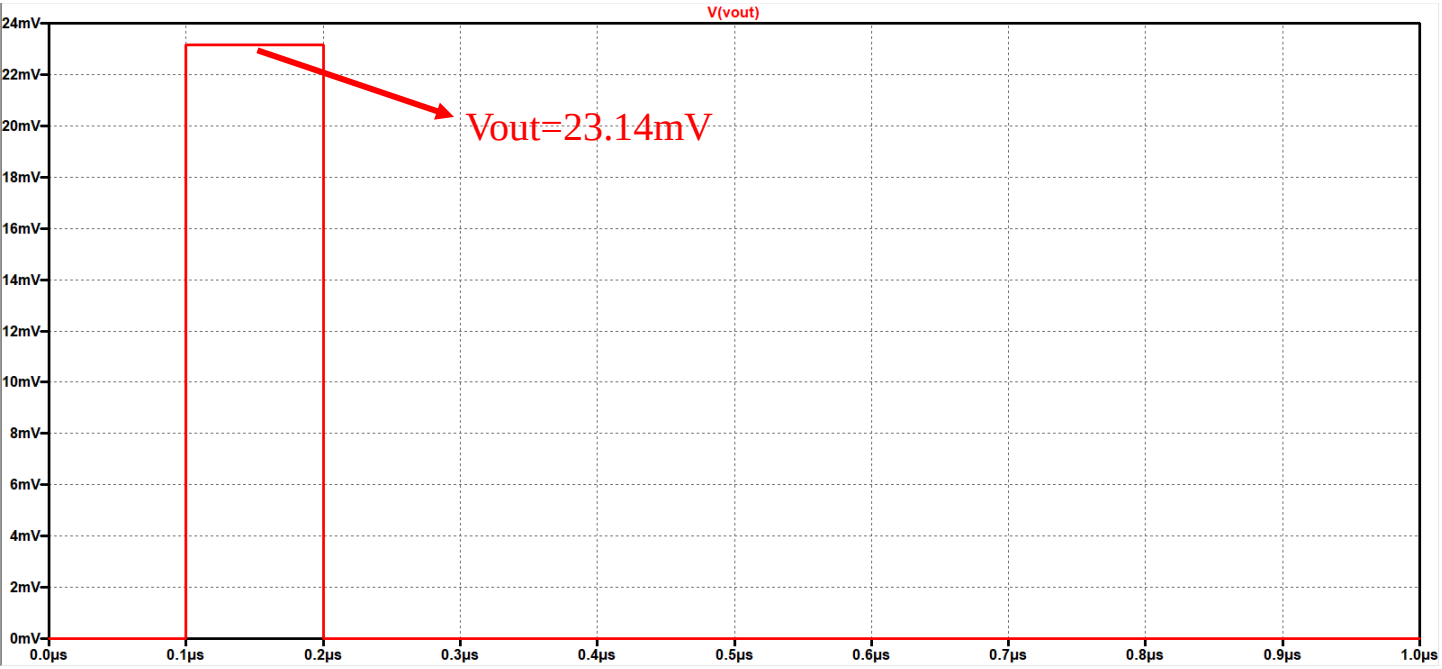
Triangular Number DAC Input 28

6-bit DAC simulation



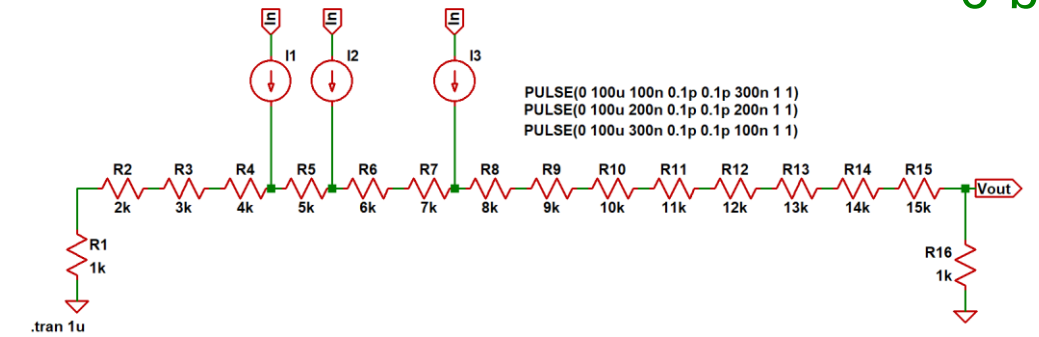
Theoretical Value Vout=28I*R/121=23.14mV

Simulation Value Vout=23.14mV



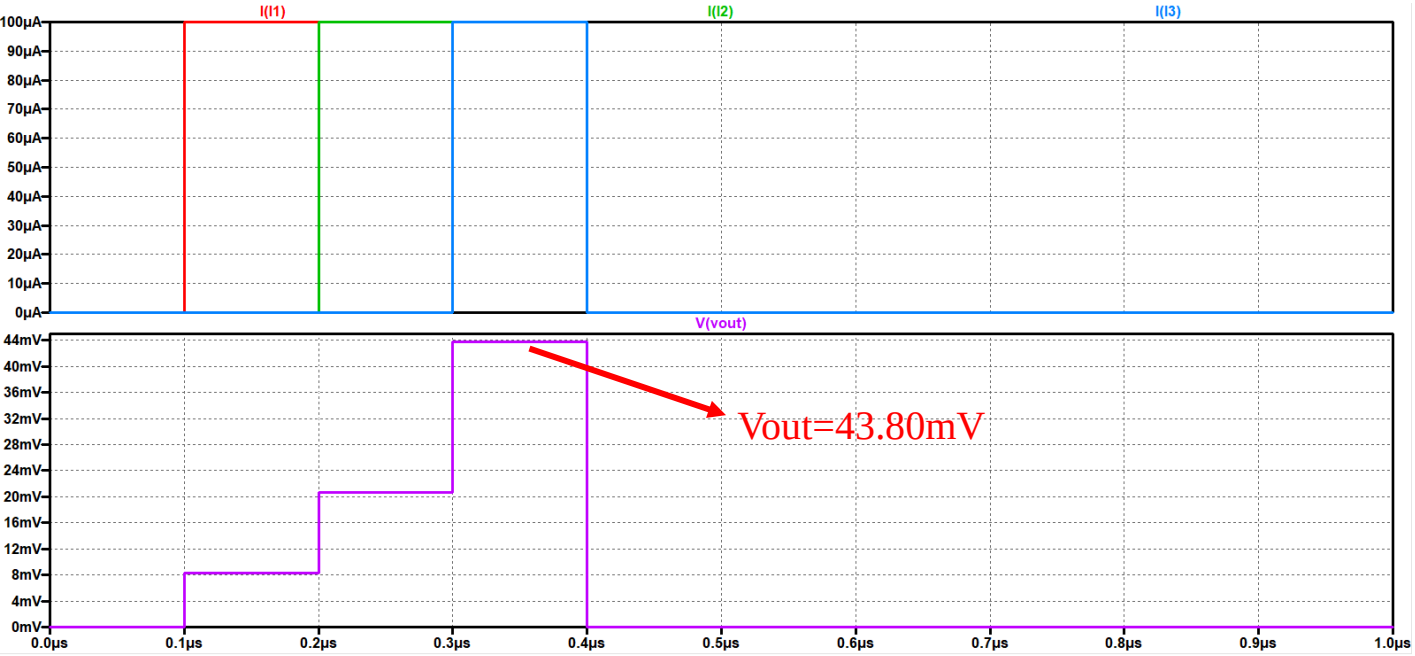
Triangular Number DAC Input 53

6-bit DAC simulation



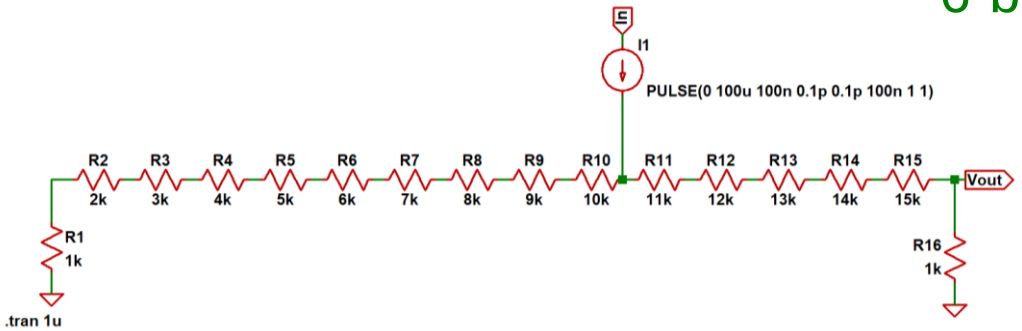
Theoretical Value Vout=53I*R/121=43.80mV

Simulation Value Vout=43.80mV



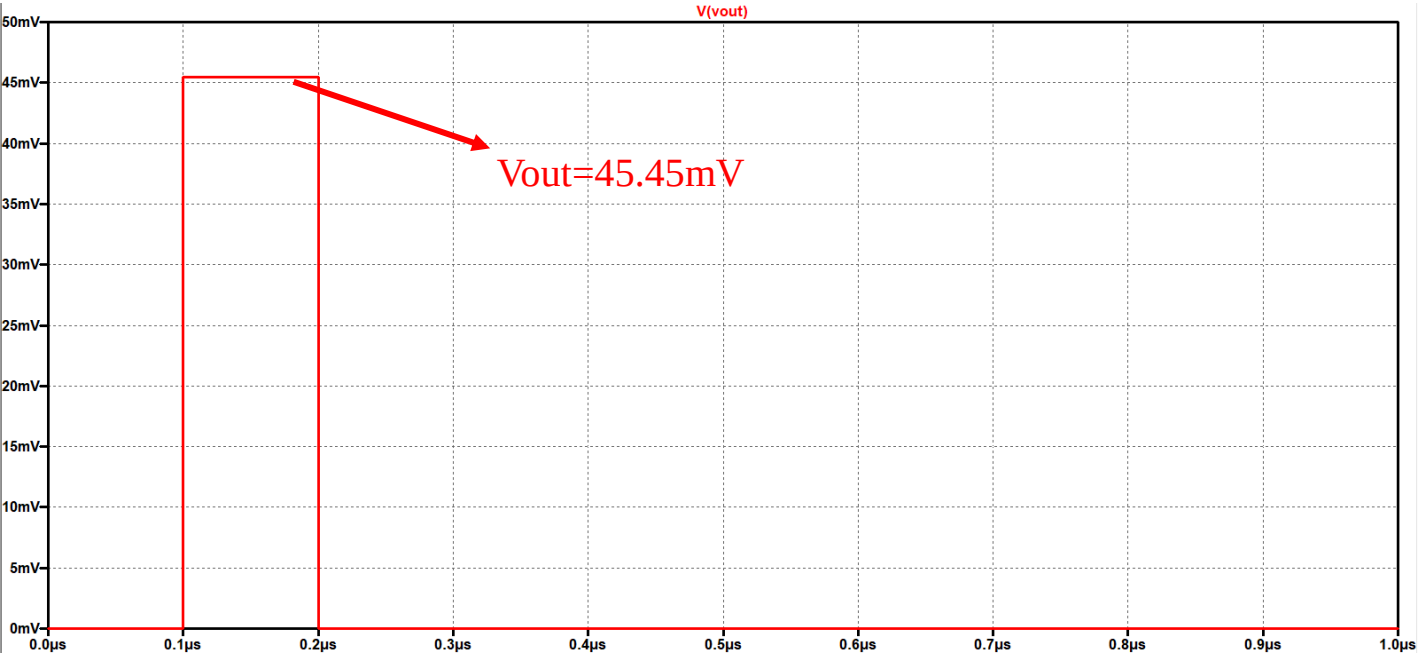
Triangular Number DAC Input 55

6-bit DAC simulation



Theoretical Value Vout=55I*R/121=45.45mV

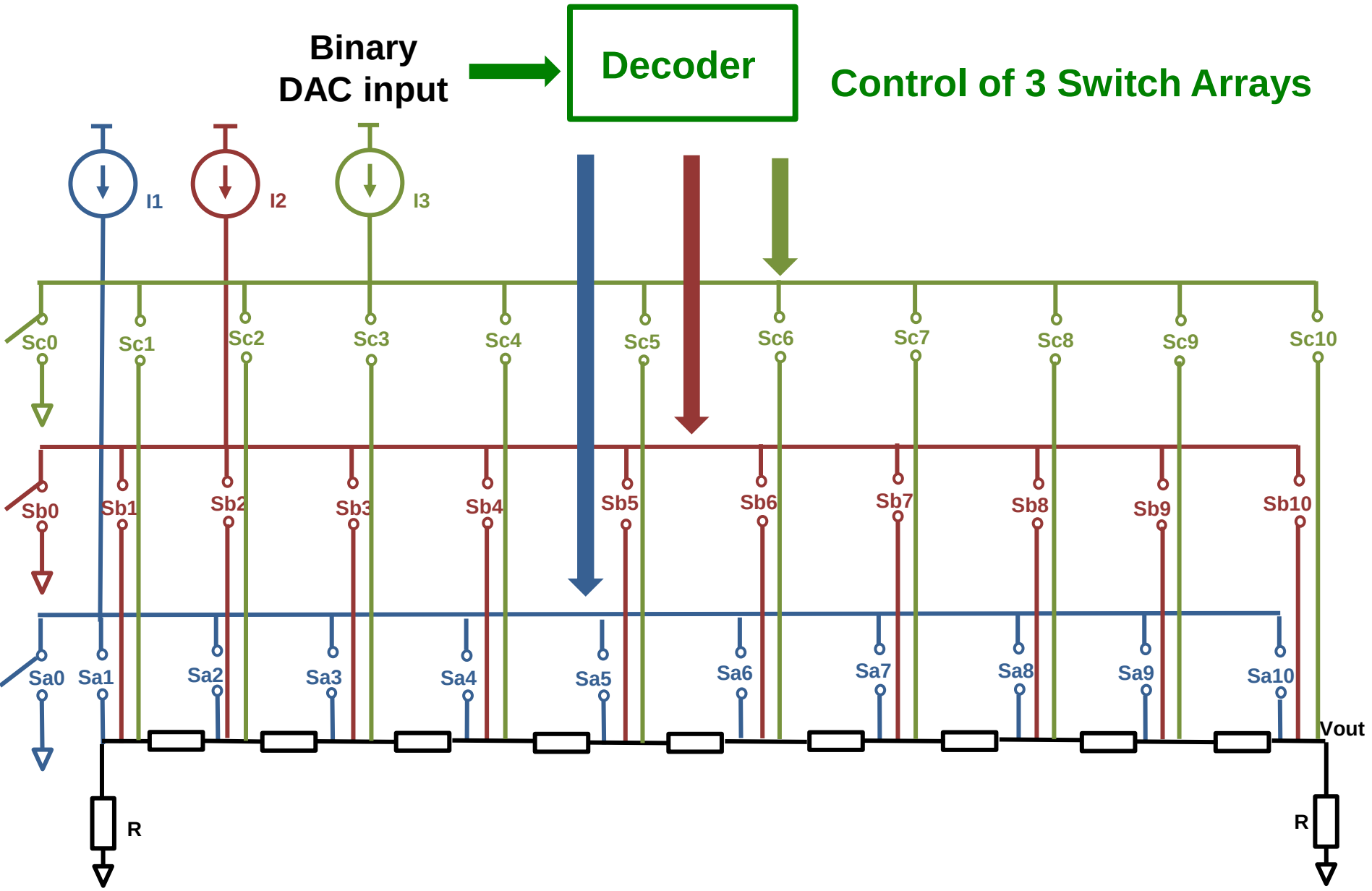
Simulation Value Vout=45.45mV



Contents

Circuits for Explanation of Switches

Sa0,... Sa10, Sb0,... Sb10, Sc0,... Sc10



Example: Digital input is 5

The digital input is 5 in decimal or B5, B4, B3, B2, B1 and B0 are 000101.
Three switches of Sa1, Sb1, and Sc2 are ON (Sa1 =1, Sb1 = 1, Sc2 = 1, 5=1+1+3) and the other switches are OFF.

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

Triangular Number DAC Decoder Check Program

Input

B5, B4, B3, B2, B1, B0(from 0 to 63)

Sa00 = Logical expression with B5, B4, B3, B2, B1, B0

Sa01 = Logical expression with B5, B4, B3, B2, B1, B0

:

Sb00 = Logical expression with B5, B4, B3, B2, B1, B0

Sb01 = Logical expression with B5, B4, B3, B2, B1, B0

:

Sc00 = Logical expression with B5, B4, B3, B2, B1, B0

Sc01 = Logical expression with B5, B4, B3, B2, B1, B0

:

DataA = 0·Sa00+1·Sa01+3·Sa02+6·Sa03+10·Sa04

+15·Sa05 +21·Sa06 +28·Sa07+36·Sa08+45·Sa09+66·Sa10.

DataB = 0·Sb00+1·Sb01+3·Sb02+6·Sb03+10·Sb04

+15·Sb05+21·Sb06+28·Sb07+36·Sb08+45·Sb09+66·Sb10.

DataC = 0·Sc00+1·Sbc01+3·Sbc02+6·Sbc03+10·Sbc04

+15·Sc05+21·Sc06+28·Sc07+36·Sc08+45·Sc09+66·Sc10.

Data = B0 + 2・B1 ＋ 4・B2 ＋8・B3 + 16・B4 + 32・B5

Triangular Number DAC Decoder Check program

Output

B5, B4, B3, B2, B1, B0

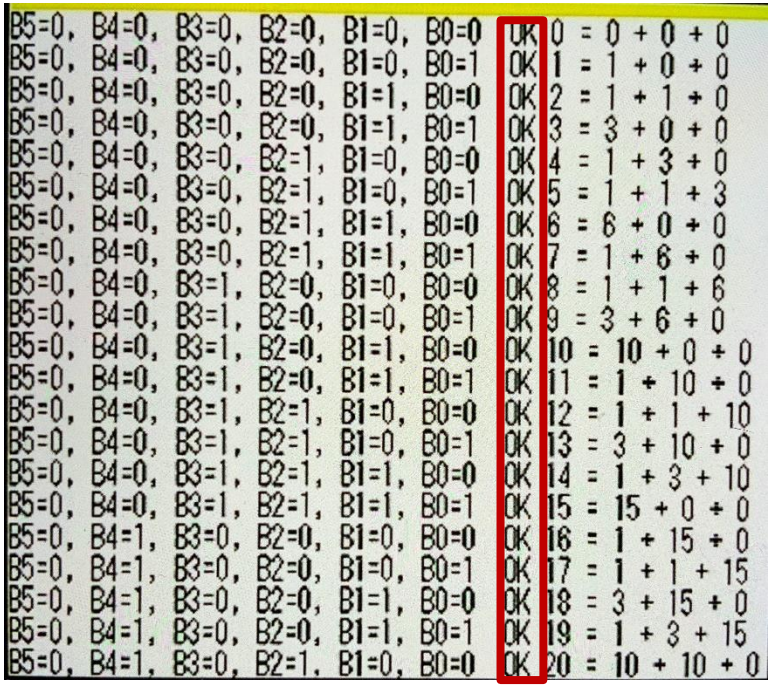
If Data = DataA + DataB + DataC,

the output is OK.

If Data ! = DataA + DataB + DataC,

the output is WRONG.

6-bit binary input　　decode



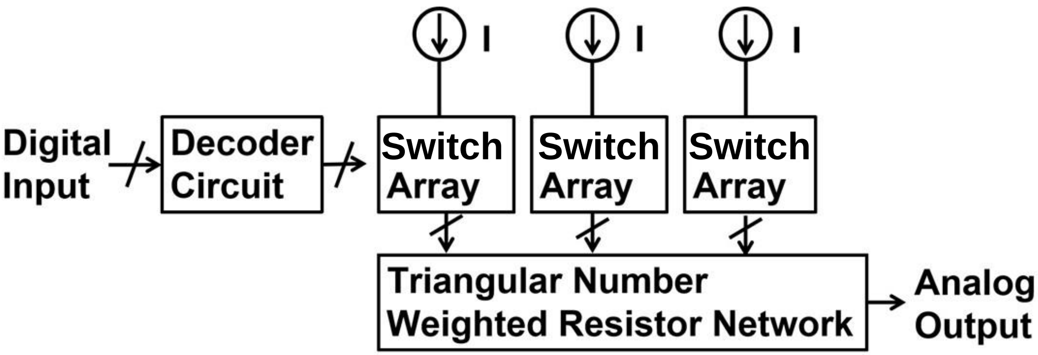
Calculation results are correct.

Contents

Summary

Triangular number DAC

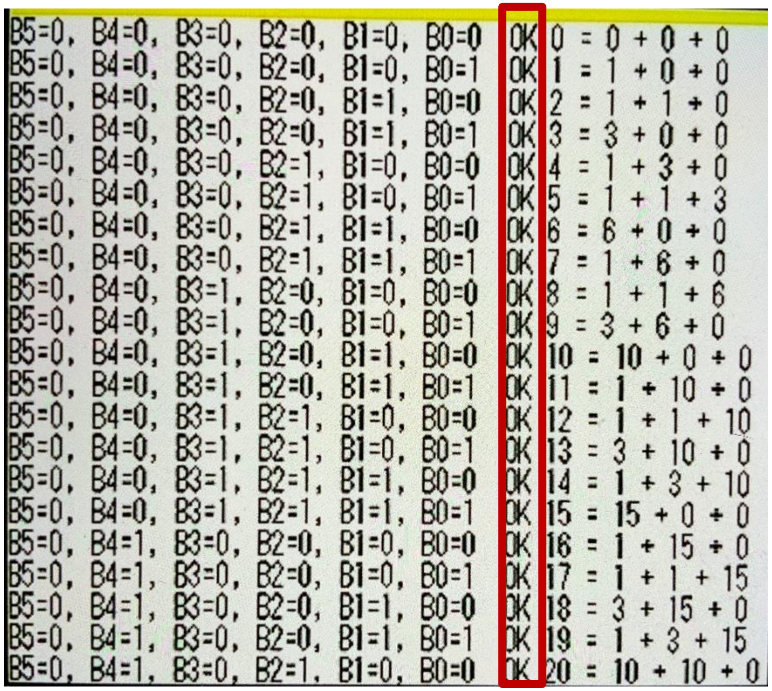
3 current sources
3 switch arrays



6-bit DAC design

6-bit binary input

decode

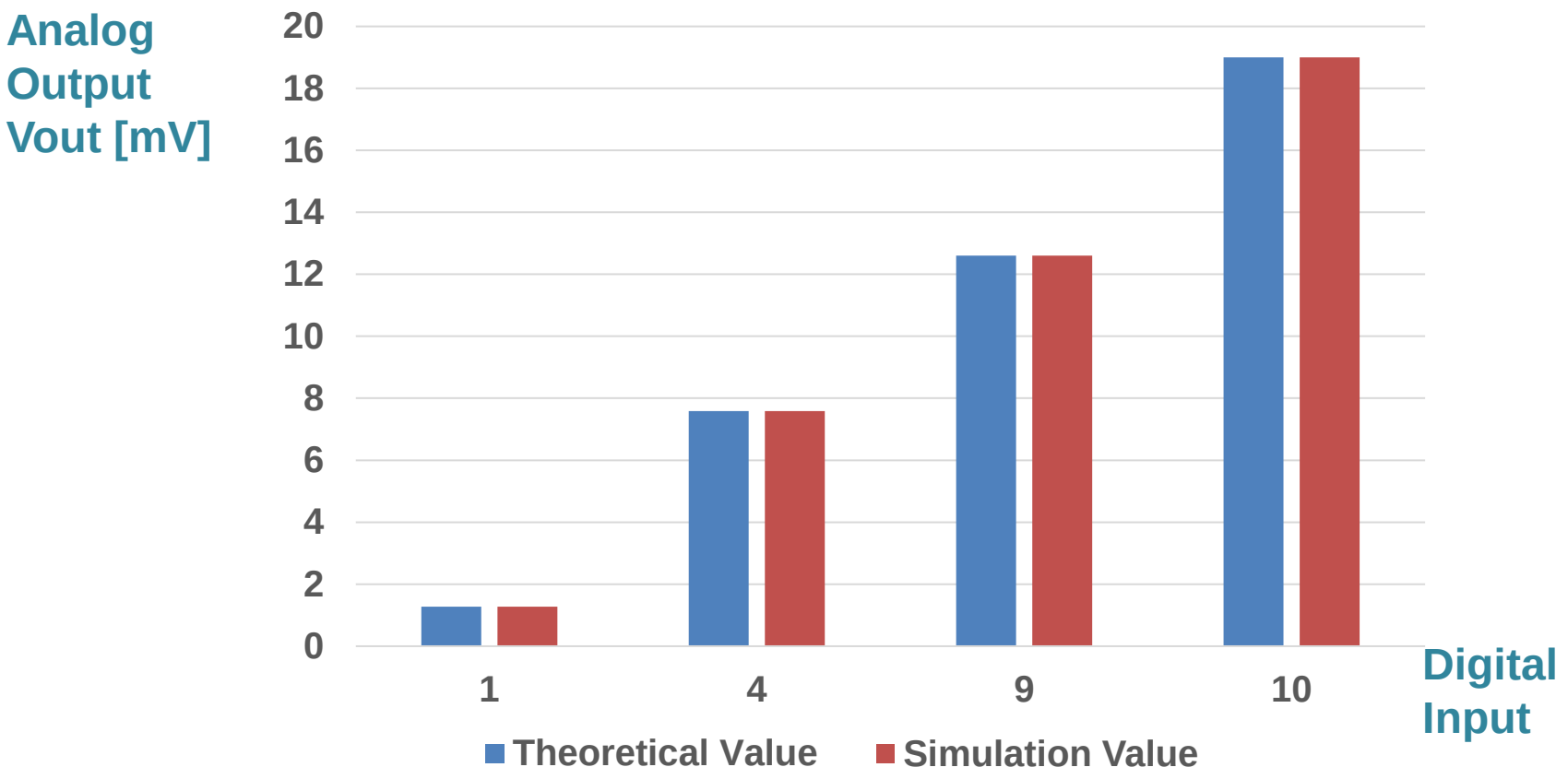


```
B5=0, B4=0, B3=0, B2=0, B1=0, B0=0 OK 0 = 0 + 0 + 0
B5=0, B4=0, B3=0, B2=0, B1=0, B0=1 OK 1 = 1 + 0 + 0
B5=0, B4=0, B3=0, B2=0, B1=1, B0=0 OK 2 = 1 + 1 + 0
B5=0, B4=0, B3=0, B2=0, B1=1, B0=1 OK 3 = 3 + 0 + 0
B5=0, B4=0, B3=0, B2=1, B1=0, B0=0 OK 4 = 1 + 3 + 0
B5=0, B4=0, B3=0, B2=1, B1=0, B0=1 OK 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
B5=0, B4=0, B3=1, B2=0, B1=1, B0=1 OK 11 = 1 + 10 + 0
B5=0, B4=0, B3=1, B2=1, B1=0, B0=0 OK 12 = 1 + 1 + 10
B5=0, B4=0, B3=1, B2=1, B1=0, B0=1 OK 13 = 3 + 10 + 0
B5=0, B4=0, B3=1, B2=1, B1=1, B0=0 OK 14 = 1 + 3 + 10
B5=0, B4=0, B3=1, B2=1, B1=1, B0=1 OK 15 = 15 + 0 + 0
B5=0, B4=1, B3=0, B2=0, B1=0, B0=0 OK 16 = 1 + 15 + 0
B5=0, B4=1, B3=0, B2=0, B1=0, B0=1 OK 17 = 1 + 1 + 15
B5=0, B4=1, B3=0, B2=0, B1=1, B0=0 OK 18 = 3 + 15 + 0
B5=0, B4=1, B3=0, B2=0, B1=1, B0=1 OK 19 = 1 + 3 + 15
B5=0, B4=1, B3=0, B2=1, B1=0, B0=0 OK 20 = 10 + 10 + 0
```

Calculation results are correct.

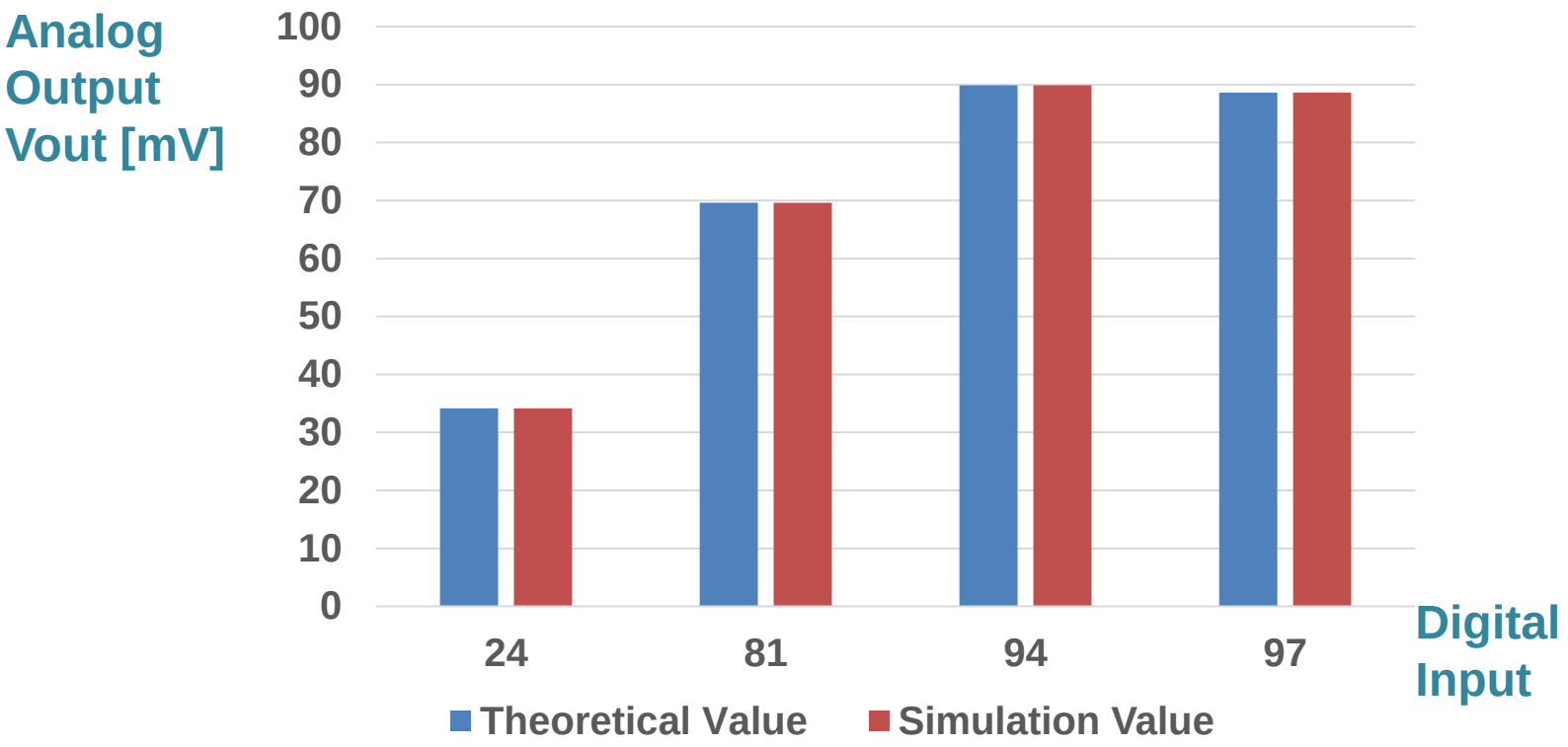
Contents

Square Numbers DAC Results



Input	1	4	9	16
Theoretical Value	Vout=I*R/79 =1.27mV	Vout=6I*R/79 =7.59mV	Vout=10I*R/79 =12.6mV	Vout=15I*R/79 =19.00mV
Simulation Value	Vout=1.27mV	Vout=7.59mV	Vout=12.6mV	Vout=19.00mV

Square Numbers DAC Results



Input	24	81	94	97
Theoretical Value	Vout=I*R/79 =34.18mV	Vout=55I*R/79 =69.62mV	Vout=71I*R/79 =89.87mV	Vout=70I*R/79 =88.61mV
Simulation Value	Vout=34.18mV	Vout=69.62mV	Vout=89.87mV	Vout=88.61mV

Contents

Conclusion

- **Completely new DAC architectures**
  based on number theory
- **Design of 6-bit DAC**
  based on triangular number theory
  → **Verified by SPICE simulation**
- **Decoder Design**
- **Decoder Check program**

Fermat polygonal number theorem
was finally proven in 1813,
by Augustin-Louis Cauchy.

1789 –1857

ご清聴ありがとうございました

Thank you for listening

謝謝

Merci de votre attention