Design of (3,2) and (4,2) CNTFET Ternary Counters for Multipliers

Original Research Article

Abstract

The reduction trees of combinational multipliers are widely applying counters. To be able to compare the ternary and the binary approaches, Nanotube Field-Effect Transistor (CNTFET) ternary (3,2) and ternary (4,2) counters have been designed. The ternary (4,2) counter is compared with the binary (7,3) counter as both compute approximately the same amount of information. The binary counter is more efficient. However, comparing counters is not enough: in the Wallace reduction tree of the ternary multiplier, there are two times more lines to reduce compared to the binary one, as a 1-trit multiplier generates both product and carry terms. Comparing the Wallace tree of an 8*8-trit multiplier and a 12*12-bit binary one also shows that the binary implementation is the most efficient.

Keywords: CNTFET; Ternary counters, Wallace trees, Combinational multipliers

2010 Mathematics Subject Classification: 68U01; 94D99;

1 Introduction

Binary counters are widely used in combinational multipliers to implement Wallace or Dadda reduction trees [1]. Fig. 1 shows how binary (3,2) counters are used to implement the Wallace tree of a 8*8 bit multiplier.

Typical binary multipliers can be decomposed in two parts: the first one generates the partial products and the second one reduces the partial products into two lines to be added by a final fast adder.

- The first part generates 8 partial products of 8 bits, i.e. 64 bits that are the products of $Ai \times Bj$ for $0 \leq i < 8$ and $0 \leq j < 8$. The binary product $Ai \times Bj$ is implemented by a And gate.
- With Wallace tree and (3,2) counters, 3 lines of partial products are reduced to two lines by using (3,2) counters and half adders. It turns out that (3,2) counters are the 1-bit full adders. For 8*8 bit multipliers, there are several steps of parallel additions of lines of partial products: 8 to 6, 6 to 4, 4 to 3, and finally 3 to 2. At that point, a fast adder is used to add

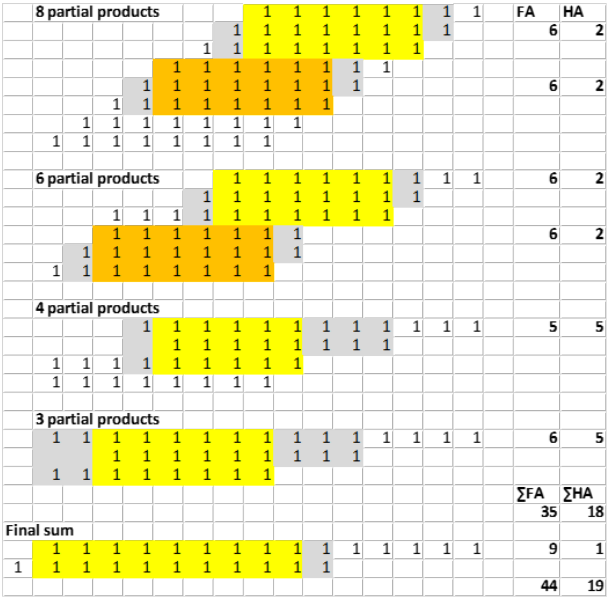

Figure 1: 8*8 bit multiplication with a Wallace tree

the two final lines. Other counters can be used, such as (4,2) or (7,3) counters to reduce the number of reduction stages.

Obviously, this technique can be used for ternary multipliers. (3,2) or (4,2) ternary counters are generally used even if larger counters can be designed [2]. Most proposals of ternary counters use capacitive theshold logic [3, 4, 5, 6]. Paper [7] proposes quaternary (4,2) and (5,2) counters. In these papers, the counters are called compressors. There is a difference between counters and compressors. (3,2) counters sum the 3 input values and generate two output values while (3,2) compressors are (3,2) counters with supplementary carry input and output.

In this paper, we propose a (4,2) ternary counter that is based on a ternary full adder using ternary carry values. The paper is organized as follow:

- The drawback of capacitive threshold logic is discussed.
- The ternary full adder with ternary carries is derived from the usual ternary full adder and its performance are discussed.
- The (4,2) ternary counter is presented with its performance.
- The (4,2) ternary counter is compared with different implementations of (7,3) binary counters that compute approximately the same amount of information
- Reduction trees of 8*8 trit multipliers with (4,2) counters and 12*12 bit multipliers with (7,3) counters are compared.
- The conclusion summarizes the results.

2 Drawbacks of capacitive theshold logic

Threshold logic is based on a linear sum of input values that is compared to a threshold value. If $\Sigma(in_i) > T$ then out=1 else out=0. As an example, Fig. 2 shows how the carry output of a (3,2)

Table 1: TFAs Comparison

TFA / Year	CNTFETs Count	Power (μW)	Max. Delay (ps)	Max. PDP (x10^{-18} J)	Max. EDP (x10^{-27} J.s)
In [8] 2011	412	1.36	88	120	10.5
In [9] 2017	105	1.13	68	77	5.2
In [10] 2017	74	0.82	146	120	17.5
In [11] 2018	98	0.16	192	31	5.9
In [12] 2018	89	0.44	48	21	1
In [13] 2019	142	4.62	94	434	40.8
In [14] 2020	106	0.13	269	35	9.4
In [15] 2020	49	1.23	192	236	45.3
In [16] 2021	337	1.96	78	153	11.9
In [16] Design 2	37	0.81	262	212	55.5
In [17] 2021	74	0.13	98	12.74	1.2
In [18] 2021	54	0.43	47	20	0.9
In [19] 2023 TFA1	59	0.46	27	12.5	0.3
In [19] TFA2	55	0.22	34	7.5	0.25

counter is implemented in [4]. The approach looks simple and thus attractive. Summing A, B, C inputs leads to 0 to 6 values to be compared to two different thresholds (2.5 and 5.5). It turns out that this approach has serious drawbacks.

- Threshold logic has been used by one digital logic family: Resistor Transistor Logic (RTL) in the 70's has been used for a while, but was quickly replaced by DTL, then TTL. All logic families after RTL have replaced a linear combination of inputs by a non-linear combination of inputs.
- Noise margins are the main issue of linear combination of inputs. In digital circuits, noises are not only random noises but include coherent noises. A typical example is the ground shift due to current through resistive ground planes, as shown in Fig.3 for two NMOS inverters. It means that for n-input gates, the same amount of static noise can be present on each input. If the noise margin for one input is $NM \leq V/2$, the noise margin for coherent noises on n inputs is NM/n. If $NM < V/2$, it means that $NM2 < V/4, NM3 < V/6\ldots$ and $NMn < V/2n$.
- Table 1 compares the performance of ternary full adders published since 2011. Paper [15] is the only one using capacitive threshold logic. While the transistor counts are low, both power, max delays, PDP and EDP are larger than for the other implementation approaches.

3 Methodology

In this paper, we present the implementations of ternary (3,2) and (4,2) CNTFET ternary counters and (7,3) binary counters. The significant figures to consider includes switching times, power dissipation, chip area, etc.

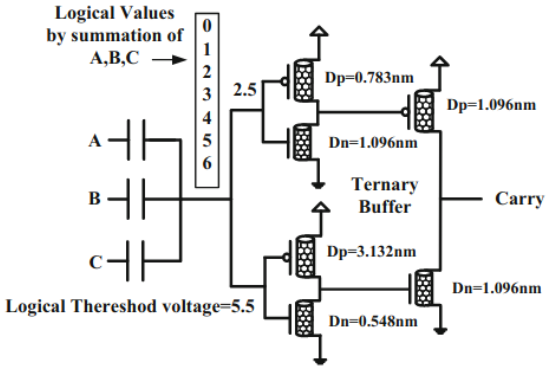

Figure 2: Carry input for 3-2 compressor design [4]

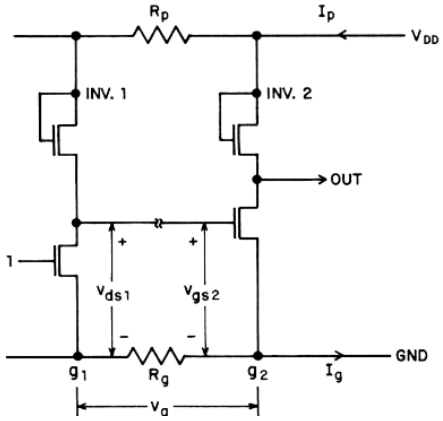

Figure 3: Resistive ground shifts

3.1 CNTFET technology

We use CNTFET technology as it is used in most papers presenting ternary or quaternary implementation of adders, multipliers, counters, etc. One advantage of CNTFET technology is that the threshold levels of gates only depend on the diameter of individual transistors, which facilitates the design of m-valued circuits. All simulations are done with the 32nm CNTFET parameters of Stanford library [20] that are used by most m-valued CNTFET designers.

3.2 Propagation delays

For all combinational circuits, the important information is the propagation delay corresponding to the critical paths. We will only present the propagation delays corresponding to these critical paths.

3.3 Power dissipation and Power-Delay Product (PDP)

Both power dissipation and PDP directly depends on the duration of the input signals. It is important to use the same input signal for all designs. For all simulations, we use the input waveforms shown in Fig. 4 for ternary circuits. We have verified that the delays for 0-2 or 2-0 ternary transitions are always less than ternary transitions 0-1, 1-2, 2-1 or 1-0. The binary waveforms used for the (7,3) binary counters are presented in Fig. 5.

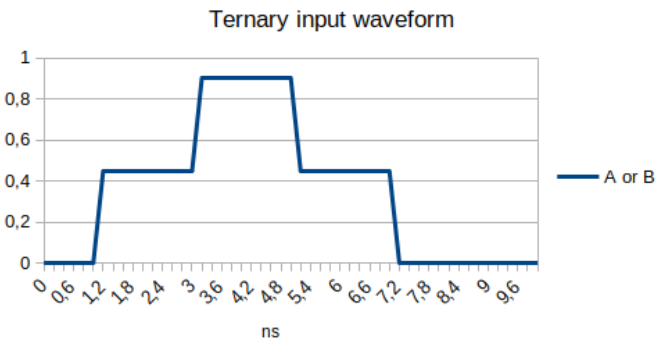

Figure 4: Ternary input and carry waveforms

3.4 Chip area

We use a rough evaluation of the chip area by summing the diameters of all the used transistors by each circuit. This rough evaluation is a little bit better than the transistor count. In this paper, we use the diameter values presented in Table 2.

3.5 Circuit styles

We only consider circuit styles with the following properties:

- No static power dissipation. This is why we only consider ternary circuits with two power supplies. Ternary circuits with only one power supply have static power dissipation for level 1.

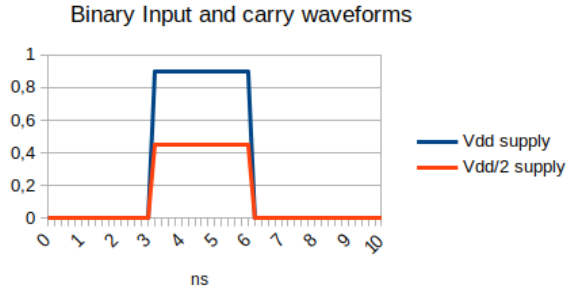

Figure 5: Binary circuit input and carry waveforms

Table 2: Transistor diameters

	n	Diameter(nm)
D1	10	0.783
D2	19	1.487
D3	29	2.270
D4	37	2.896

- The circuit outputs have full swing. Reduced swings degrade noise margins and can degrade the operation of cascaded circuits, such as Carry Propagate Adders (CPAs).
- The circuits should have a sufficient driving capability.
- As previously justified, we exclude threshold logic gates.

4 A (3,2) ternary counter

4.1 Design of the counter

The ternary (4,2) counter will be based on a (3,2) ternary counter, that corresponds to a ternary adder with ternary carry values.

Only considering columns C_0 and C_1 of Table 3 corresponds to the truth table of a 1-trit ternary full adder, for which carry values are only 0 and 1. The corresponding implementations and performance are detailed in [21]. The (3,2) ternary counter uses the MUX approach:

When C_{in}=0

- When B=0 then Sum=A
- When B=1 then Sum = (A+1) mod(3) quoted as A^1
- When B=2 then Sum = (A+2) mod(3) quoted as A^2
- When B=0 then C_{out}=0
- When B=1 then C_{out}=1 when $A = 2$ else 0
- When B=2 then C_{out}=1 when $A > 0$ else 0

When C_{in}=1

- When B=0 then Sum=A^1

Table 3: Truth table of a ternary full adder with ternary carries

C_{in}=0				C_{in}=1				C_{in}=2			
a	b	S_0	$C_{out}0$	a	b	s1	$C_{out}1$	a	b	S_0	$C_{out}0$
0	0	0	0	0	0	1	0	0	0	2	0
0	1	1	0	0	1	2	0	0	1	0	1
0	2	2	0	0	2	0	1	0	2	1	1
1	0	1	0	1	0	2	0	1	0	0	1
1	1	2	0	1	1	0	1	1	1	1	1
1	2	0	1	1	2	1	1	1	2	2	1
2	0	2	0	2	0	0	1	2	0	1	1
2	1	0	1	2	1	1	1	2	1	2	1
2	2	1	1	2	2	2	1	2	2	0	2

- When B=1 then Sum=A^2
- When B=2 then Sum= A
- When B=0 then C_{out}=1 when $A = 2$ else 0
- When B=1 then C_{out}=1 when $A > 0$ else 0
- When B=2 then C_{out}=1

When C_{in}=2

- When B=0 then Sum=A^2
- When B=1 then Sum=A
- When B=2 then Sum= A^1
- When B=0 then C_{out}=0 when $A = 0$ else 1
- When B=1 then C_{out}=1
- When B=2 then C_{out}=1 when $A < 2$ else 2

Circuits implementing A^1 and A^2 are shown in Fig. 6. The MUX3 circuit is shown in Fig. 7 The overall (3,2) counter is presented in Fig.8.

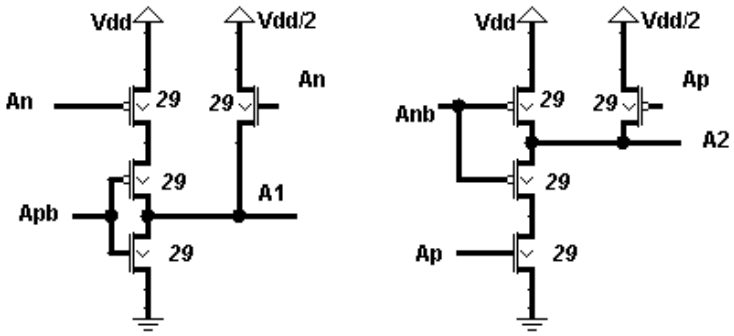

Figure 6: A^1 and A^2 circuits

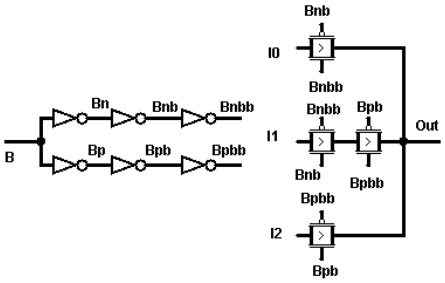

Figure 7: 3-input MUX with ternary control

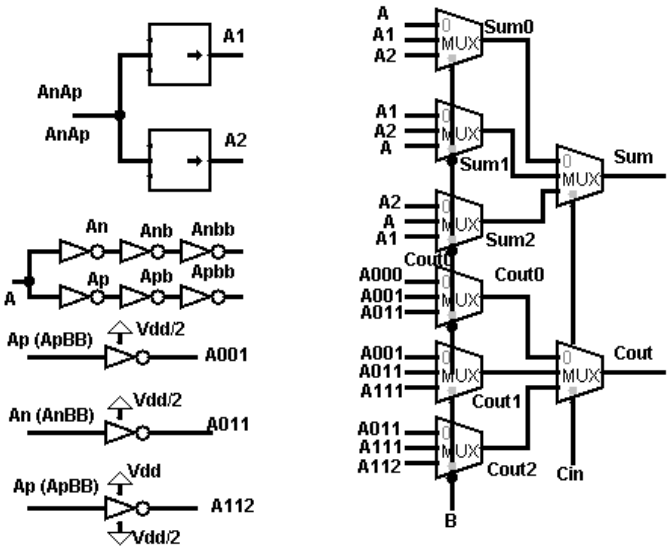

Figure 8: Ternary (3,2) counter

Table 4: Worst case delays according to inputs of the (3,2) counter (CL=2 fF)

	ts (ps)	tc(ps)
Input A	72	68
Input B	90	89
Input C_{in}	59	58

4.2 Performance of the (3,2) counter

As symmetries are present in the counter (Fig. 8), only three propagation delays should be considered:

- A to Sum and C_{out}.
- B to Sum and C_{out}.
- C_{in} to Sum and C_{out}.

According to Table 4, the critical delay corresponds to Input B.

The worst case delay and the power dissipation according to CL are presented in Fig. 9.

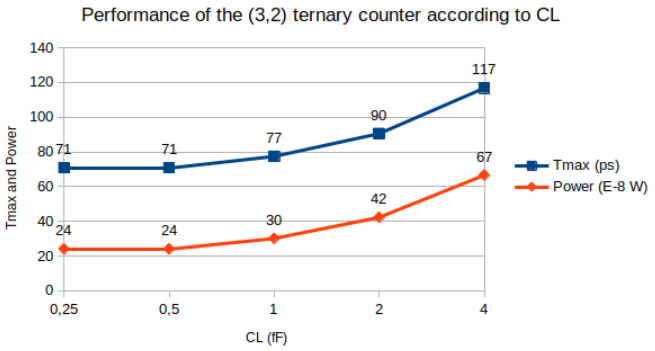

Figure 9: Worst case delay and Power according to CL for the (3,2) counter

5 A ternary (4,2) counter

5.1 Design of the (4,2) counter

Table 5 shows how S_0 and C_0 of a (3,2) counter with X_2,X_3 and X_4 inputs and S_0 and C_0 outputs can be combined with X_1 output to get the Sum and C_{out} outputs of the (4,2) counter.

Sum = $(S_0 + X_1)$ (mod 3). S_0, S_0^1 and S_0^2 are the inputs of a ternary 3-input MUX controlled by X_1. Similarly, one MUX switches the correct carry values according to S_0 when C_0=0 and a second one when C_0=1. When C_0=2, C_{out} =2. A final MUX switches the correct C_{out} value according to C_0.

Table 5: Truth Table of a (4,2) ternary counter

X_2+X_3+X_4	S_0	C_0	X_1	X_1+X_2+X_3+X_4	Sum	C_{out}
0	0	0	0/1/2	0/1/2	0/1/2	0/0/0
1	1	0	0/1/2	1/2/3	1/2/0	0/0/1
2	2	0	0/1/2	2/3/4	2/0/1	0/1/1
3	0	1	0/1/2	3/4/5	0/1/2	1/1/1
4	1	1	0/1/2	4/5/6	1/2/0	1/1/2
5	2	1	0/1/2	5/6/7	2/0/1	1/2/2
6	0	2	0/1/2	6/7/8	0/1/2	2/2/2

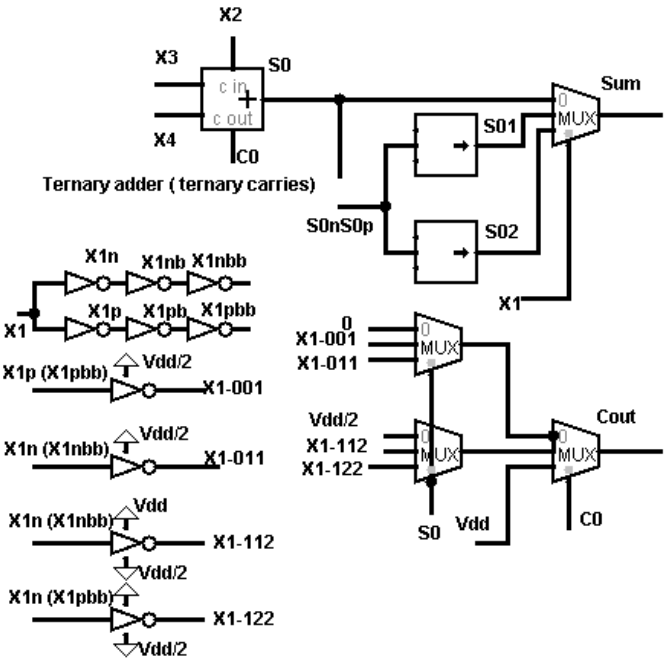

Figure 10: Ternary (4,2) counter

5.2 Performance of the (4,2) counter

A critical delay path is obtained when X_1=0, X_2=0/1/2/1/0, X_3=1 and X_4=1.Then Sum = 2/0/1/0/2 and C_{out}=0/1/1/1/A critical delay path is obtained when X_1=0, X_2=0/1/2/1/0, X_3=1 and X_4=1; Then Sum = 2/0/1/0/2 and C_{out}=0/1/1/1/0. X_2 signal has been shown in Fig. 4.

Fig. 11 presents the max delay and power dissipation according to CL capacitive load. PDP is the product max delay * Power.

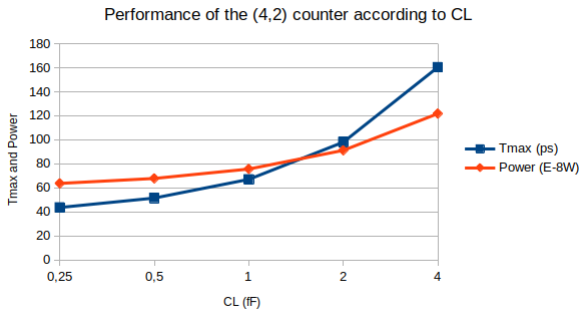

Figure 11: Performance according to CL for the (4,2) counter

6 A (7,3) CNTFET binary counter

6.1 Design

The 1-bit full adder is also the (3,2) binary counter. A (7,3) binary counter can be build using three (3,2) counters as shown in Fig. 12. The final implementation depends on the used 1-bit adder. We consider two different 1-bit adders. The first one is a 14T full adder presented in Fig. 13. It corresponds to the following equations:

- Sum = a $\oplus$ b $\oplus$ c
- If a $\oplus$ b = 1 then $C_{out} = C_{in}$ else C_{out} = a

The second one is the typical 28T 1-bit adder (Fig. 14).

While the ternary circuits use a V_{dd} power supply to operate the three different levels, the binary circuits can operate either with V_{dd} or $V_{dd}/2$ power supplies. In the second case, the dynamic power dissipation is roughly divided by 4.

A detailed examination of the performance of the 14T binary adder can be found in [21]. The 28T performance is detailed in [22].

6.2 Performance of the different (7,3) binary counters

. As shown in Fig. 12, a critical delay path is obtained when X_0=0, X_1=0, X_2=0, X_3=0, X_4=0, X_5=0/1/0, X6=1. Then out0=0/1/0 and out1 =1/0/1. X_5 signal has been shown in Fig. 5.

Fig. 11 presents the max delay and power dissipation according to CL capacitive load. PDP is the product max delay * Power.

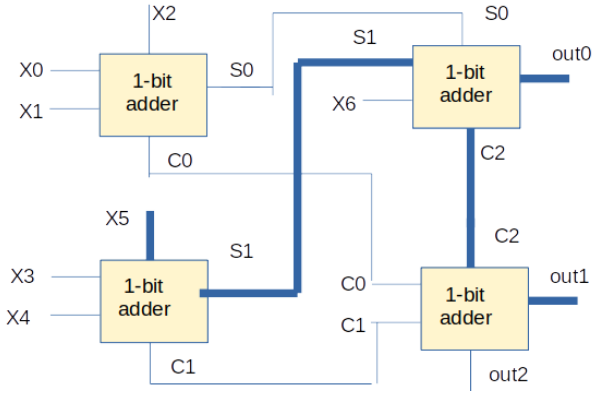

Figure 12: Binary (7,3) counter

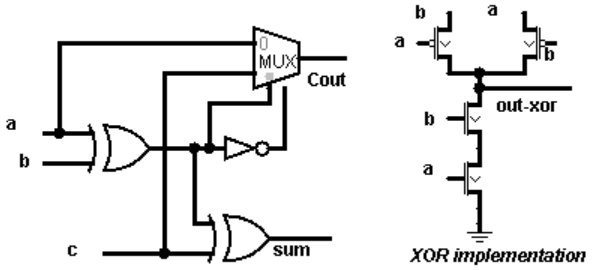

Figure 13: 14T full adder

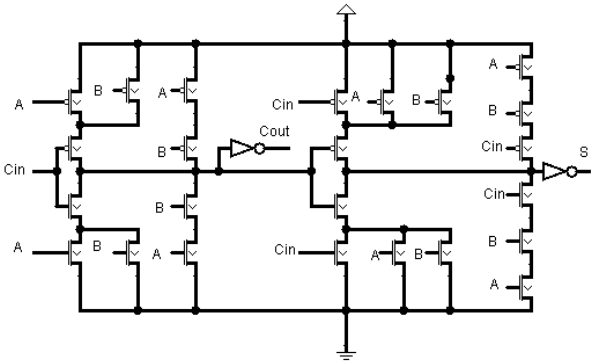

Figure 14: 28T full adder

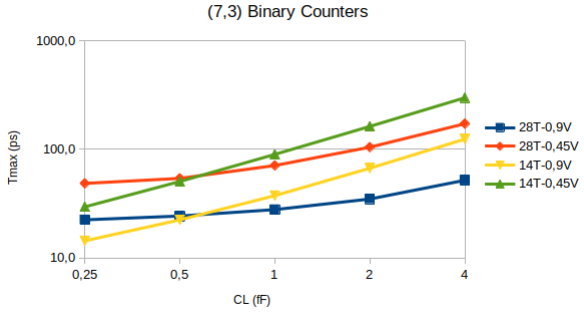

Figure 15: Tmax according to CL for (7,3) counters

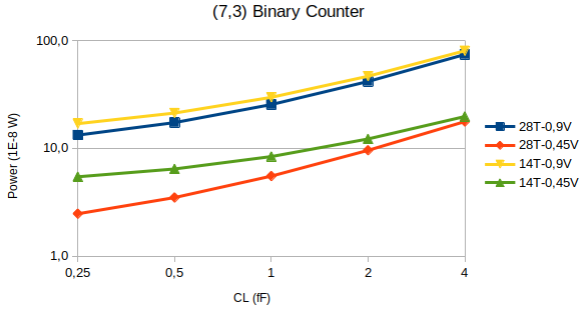

Figure 16: Power according to CL for (7,3) counters

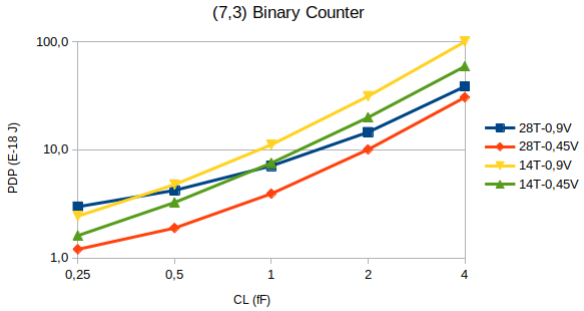

Figure 17: PDP according to CL for (7,3) counters

7 Comparing performance of the ternary (4,2) counter and binary (7,3) counters

(4,2) ternary counters compute $log_2(9) = 3{,}17$ bits of information while (7,3) binary counters compute $log_2(8) = 3$ bits. They are quite equivalent.

Fig. 18 present the maximal delay, the power dissipation, the PDP and the estimated chip area of the (4,2) ternary counter and 4 different versions of the (7,3) binary counter that use 28T or 14T full adders and 0,9V or 0,45V power supplies. Fig. 19 presents the same results as a ratio Binary performance/Ternary Performace. Ternary is better when $Ratio > 1$.

Fig. 19 facilitates the comparison. The ternary counter has smaller max delay than the binary one with 14T and 0.45V supply. The ternary counter has slightly smaller delay than the binary one with 28T and 0,45V supply and a smaller delay than the 14T with 0.45 V supply. However, the ternary counter is outperformed by the binary ones all the other features. Compared to the best binary one, the ternary Power and PDP are x9 greater for the ternary counter. Ternary chip area is x1.6 to x2.1 larger than the binary chip areas.

These results are consistent with the results of the comparison of binary and ternary full adders presented in [22] and binary and quaternary multipliers [23].

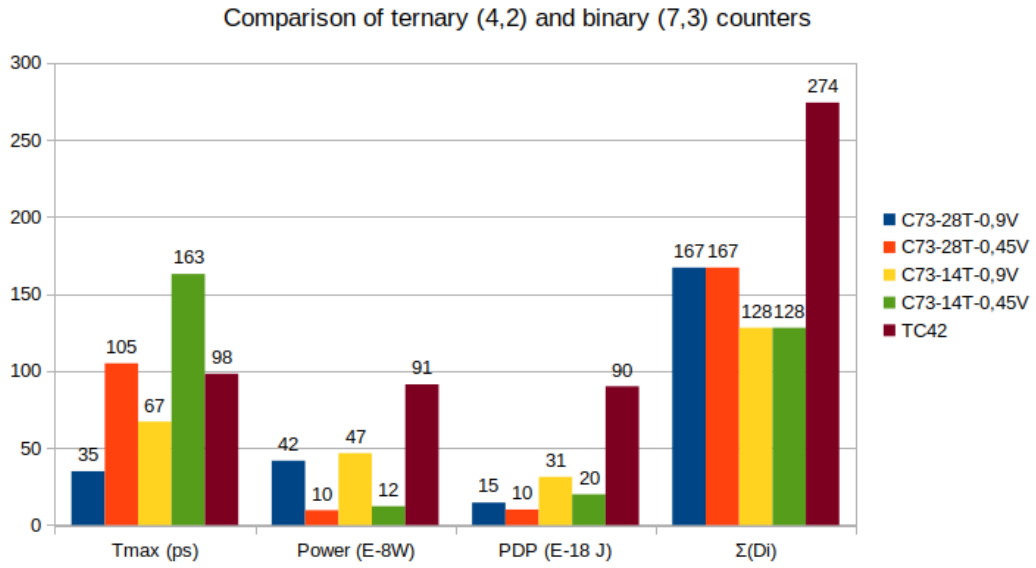

Figure 18: Performance of (4,2) ternary counters and (7,3) binary ones

8 Counters and reduction trees of multipliers

Comparing ternary counters and binary counters computing the same amount of information is not sufficient to compare the performance of ternary and binary multipliers. While R=log(3)/log(2)=1.585 is the information ratio between ternary and binary circuits,

- A n*n bit multiplier is roughly equivalent to a m*m trit multiplier with m $\approx n/R$

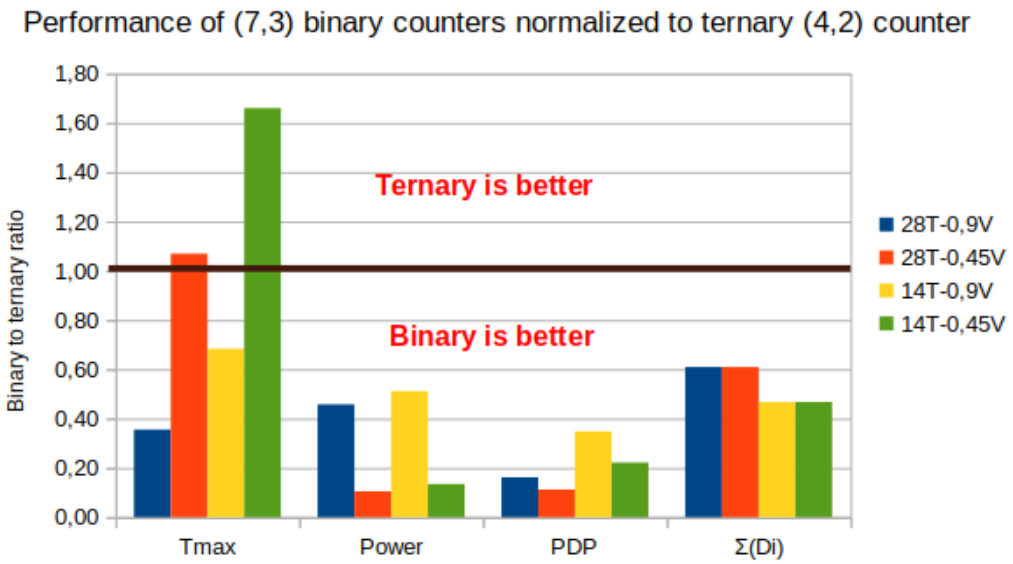

Figure 19: Performance of (7,3) binary counters normalized according to performance of (4,2) ternary counter

- There are m^2 1-trit multipliers instead of n^2 1-bit multipliers, with $m^2 = n^2/R^2 \approx n^2/2.5$. But the 1-bit multiplier is just a AND gate while the 1-trit multiplier generates both a product and a carry term (2*2=11)
- There are 2m lines of ternary products/carries of m terms while there are n lines of binary partial product of n terms.

Without going into all details, we present the Wallace tree for 8*8 trit multipliers using (4,2), (3,2) ternary counters and ternary half adders in Fig. 20. We present the corresponding reduction tree for a 12*12 bit multiplier using (7,3), (3,2) binary counters and binary half adders in Fig. 21. Table 6 shows the number of 1-trit or 1-bit multipliers, ternary C(4,2) an(3,2) counters, binary C(7,3) and C(3,2) counters for a 8*8 trit multiplier and a 12*12 bit multiplier:

- The number of 1-trit multiplier is less than 2.5 times the number of 1-bit multipliers (R^2). However, the 1-trit multiplier is far more complicated than a AND gate.
- There are less (7,3) binary counters than (4,2) ternary counters while the binary counters are faster and use less chip area (Fig. 18.
- The number of ternary (3,2) counters is slightly less than the number of binary (3,2) counters, but they are far more complicated.

The increased complexity of the ternary counters versus the binary ones is not compensated by some advantage of using these counters in the reduction trees of the corresponding multipliers.

9 Concluding remarks

We have presented a CNTFET (3,2) counter that is a ternary full adder with ternary carry values. Then a ternary (4,2) counter has been derived from the (3,2) counter together with their

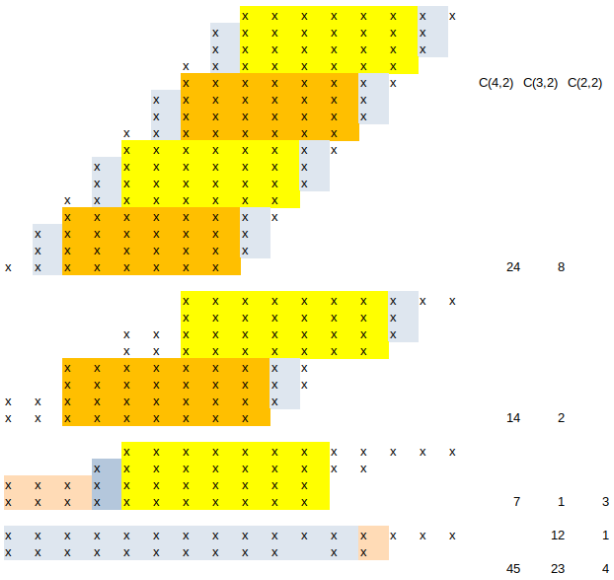

Figure 20: Wallace tree reduction for a 8*8 trit multiplier with (4,2) and (3,2) ternary counters

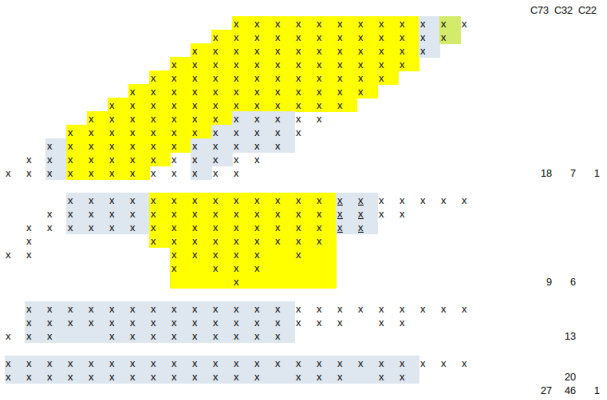

Figure 21: Wallace tree reduction for a 12*12 bit multiplier with (7,3) and (3,2) binary counters

Table 6: Comparison for 12*12 bit and 8*8 trit multipliers

	Binary	Ternary	Binary/Ternary
Ai*Bi	144	64	2.25
TC(4,2)		48	
TC(3,2)		23	
THA		4	
BC(7,3)	27		
BC(3,2)	26		
BHA	1		

performance. The ternary (4,2) counter has been compared with binary (7,3) counters using either 14 or 28 T full adders and using either 0.9 V and 0.45 V power supplies. The ternary counter is outperformed by the binary ones for most features (delay, power, PDP, chip area).

Counters are only one element in the design of reduction trees of combinational multipliers. The number of 1-bit or 1-trit multipliers, the number of lines to be reduced are also important. The ternary reduction tree suffers from the number of lines to reduce as a ternary 1-trit multiplier generates both a product and a carry. The reduction tree of a 8*8 trit multiplier and a 12*12 bit multiplier have been presented to evaluate the impact of the 1-digit multipliers and the number of lines to be reduced. The ternary reduction tree is also less efficient than the binary one.

For combinational multipliers, the ternary approach doesn't deliver any advantage versus the binary one, as both the counters and the reduction trees are less efficient according to speed, chip area and power dissipation.

References

[1] W. J. Townsend, E. E. Swartzlander Jr. and J. A. Abraham, "A comparison of Dadda and Wallace multiplier delays", Proc. SPIE 5205, Advanced Signal Processing Algorithms, Architectures, and Implementations XIII, (24 December 2003); https://doi.org/10.1117/12.507012

[2] A. G. Asibelagh, R. F. Mirzaee, "Applicability of Partial Ternary Full Adder in Ternary Arithmetic Units", arXiv preprint arXiv:1902.06742, 2019 https://arxiv.org/pdf/1902.06742.pdf

[3] E. Alkaldi, K. Navi, F. Sharifi and M.H Moayeri, "An Ultra High-Speed (4; 2) Compressor with a New Design Approach for Nanotechnology Based on the Multi-Input Majority Function", in Journal of Computational and Theorical Nanosciences, pp 1691-1696, 11, 2014?

[4] S. Tabrizchi, H. Sharifi, F. Sharifi and K. Navi, "A Novel Design Approach for Ternary Compressor Cells based on CNTFET', Circuits, Systems, and Signal Processing, 2016

[5] R.F. Mirzaee and A.Reza, "High Performance Ternary (4:2) compressor Based on Capacitive Threshold Logic", in International Journal of Electronics and Telecommunications, Vol 63, N° 4, pp 355-361.

[6] T. Sharma,L. Kumre, "Efficient Ternary Compressor Design Using Capacitive Threshold Logic in CNTFET Technology", IETE Journal of Research, 69:3, (2023) pp 1539-1549, DOI: 10.1080/03772063.2020.1871422

[7] F. Sharifi, A. Panhi, H. Sharifi, K. Navi, N. Bagherzadeh, H. Thapliyal, "Design of quaternary 4-2 and 5-2 compressors for nanotechnology", in Computer & Electronic Engineering, Vol 56, Nov. 2016, pp 64-74.

[8] S. Lin, Y.-B. Kim, and F. Lombardi, "CNTFET-Based Design of Ternary Logic Gates and Arithmetic Circuits', IEEE Trans. Nanotechnology, vol. 10, no. 2, pp. 217–225, Mar. 2011, doi: 10.1109/TNANO.2009.2036845.

[9] B. Srinivasu and K. Sridharan, "A Synthesis Methodology for Ternary Logic Circuits in Emerging Device Technologies', IEEE Trans. Circuits Syst. I, vol. 64, no. 8, pp. 2146–2159, Aug. 2017, doi: 10.1109/TCSI.2017.2686446.

[10] S. Tabrizchi, A. Panahi, F. Sharifi, K. Navi, and N. Bagherzadeh, "Method for designing ternary adder cells based on CNFETs", IET Circuits, Devices & Systems, vol. 11, no. 5, pp. 465–470, Sep. 2017, doi: 10.1049/iet-cds.2016.0443.

[11] C. Vudadha, A. Surya, S. Agrawal, and M. B. Srinivas, "Synthesis of Ternary Logic Circuits Using 2:1 Multiplexers', IEEE Trans. Circuits Syst. I, vol. 65, no. 12, pp. 4313–4325, Dec. 2018, doi: 10.1109/TCSI.2018.2838258.

[12] E. Shahrom and S. A. Hosseini, "A new low power multiplexer based ternary multiplier using CNTFETs', AEU - International Journal of Electronics and Communications, vol. 93, pp. 191–207, Sep. 2018, doi: 10.1016/j.aeue.2018.06.011.

[13] T. Sharma and L. Kumre, "CNTFET-Based Design of Ternary Arithmetic Modules', Circuits Syst Signal Process, vol. 38, no. 10, pp. 4640–4666, Oct. 2019, doi: 10.1007/s00034-019-01070-9.

[14] S. Kim, S. -Y. Lee, S. Park, K. R. Kim and S. Kang, "A Logic Synthesis Methodology for Low-Power Ternary Logic Circuits," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 67, no. 9, pp. 3138-3151, Sept. 2020, doi: 10.1109/TCSI.2020.2990748.

[15] I. Mahmoudi Salehabad, K. Navi, and M. Hosseinzadeh, "Two novel inverter-based ternary full adder cells using CNFETs for energy-efficient applications', International Journal of Electronics, vol. 107, no. 1, pp. 82–98, Jan. 2020, doi: 10.1080/00207217.2019.1636306.

[16] F. Zahoor et al., "Carbon Nanotube Field Effect Transistor (CNTFET) and Resistive Random Access Memory (RRAM) Based Ternary Combinational Logic Circuits', Electronics, vol. 10, no. 1, p. 79, Jan. 2021, doi: 10.3390/electronics10010079.

[17] F. Mahboob Sardroudi, M. Habibi, and M. H. Moaiyeri, "A low-power dynamic ternary full adder using carbon nanotube field-effect transistors', AEU - International Journal of Electronics and Communications, vol. 131, p. 153600, Mar. 2021, doi: 10.1016/j.aeue.2020.153600.

[18] S. A. Hosseini and S. Etezadi, "A Novel Low-Complexity and Energy-Efficient Ternary Full Adder in Nanoelectronics', Circuits Syst Signal Process, vol. 40, no. 3, pp. 1314–1332, Mar. 2021, doi: 10.1007/s00034-020-01519-2.

[19] R.A. Jaber, A.M. Haidar,A. Kassem, F. Zahoor "Ternary Full Adder Designs Employing Unary Operators and Ternary Multiplexers". Micromachines 2023, 14, 1064. https://doi.org/10.3390/mi14051064

[20] J. Deng and H. -. P. Wong, "A Compact SPICE Model for Carbon-Nanotube Field-Effect Transistors Including Nonidealities and Its Application—Part II: Full Device Model and Circuit Performance Benchmarking," in IEEE Transactions on Electron Devices, vol. 54, no. 12, pp. 3195-3205, Dec. 2007. https://doi.org/10.1109/TED.2007.909043.

[21] D. Etiemble, "Post algebras and ternary adders". Journal of Electrical Systems and Inf Technol 10, 20 (2023). https://doi.org/10.1186/s43067-023-00088-z

[22] D.Etiemble, "Ternary and Quaternary CNTFET Full Adders are less efficient than the Binary Ones for Carry-Propagate Adders", arXiv preprint arXiv:2207.04839, 2022, https://arxiv.org/pdf/2207.04839.pdf

[23] D.Etiemble, "CNTFET quaternary multipliers are less efficient than the corresponding binary ones", arXiv preprint arXiv:2206.03252, 2022, https://arxiv.org/abs/2206.03252