

# Сумматор Склянского, Radix-4, 8-ми битный, медленный и быстрый

## Sklansky Radix-4

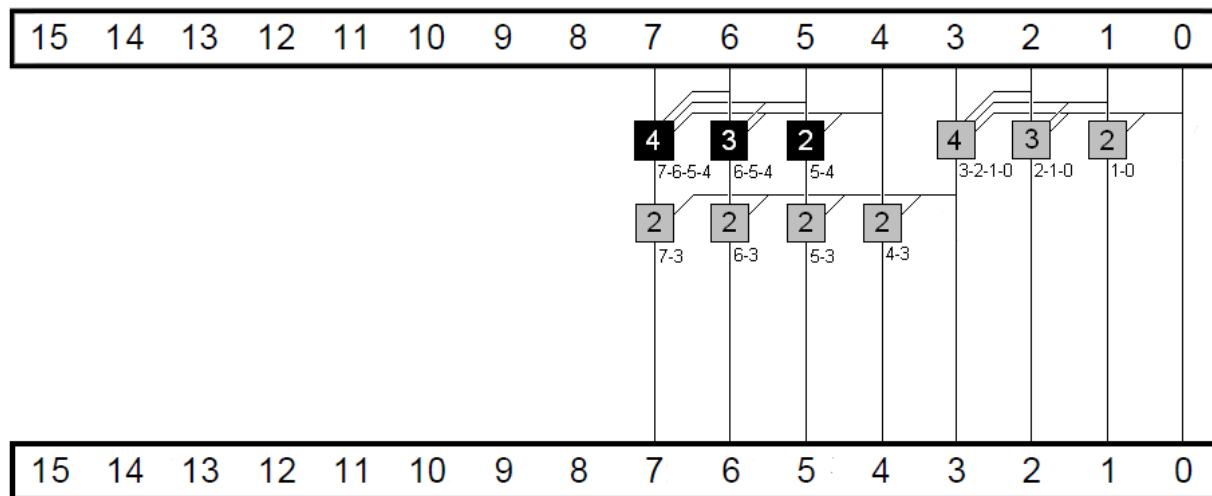


Рис.1. Граф генератора переносов сумматора Склянского, Radix-4, 8-ми разрядного (используются и двух, трёх и четырёх входные (аргументные) операторы (функции) 2G, 3G и 4G (Gray2, Gray3 и Gray4) (без вычисления функций 2P, 3P и 4P) и 2GP, 3GP и 4GP (Black2, Black3 и Black4) (с вычислением функций 2P, 3P и 4P)). Порядок подключения входов имеет значение.

Таблица валентностей операторов G и P:

Sklansky, Radix-4, 8-bit, 6dt

7	6	5	4	3	2	1	0	Время быстрого			
-----								TTL	KEY		
								Step 0,	1dt	1dt	
-----											
4	3	2	—	4	3	2	—	Step 1,	3dt.	2dt	valency-1,2,3,4
2	2	2	2	—	—	—	—	Step 2,	5dt,	3dt	valency-1,2
-----											
								Sum,	6dt	4dt	
-----											

## Медленный (с последовательными операторами G) сумматор Склянского, Radix-4, 8-ми битный

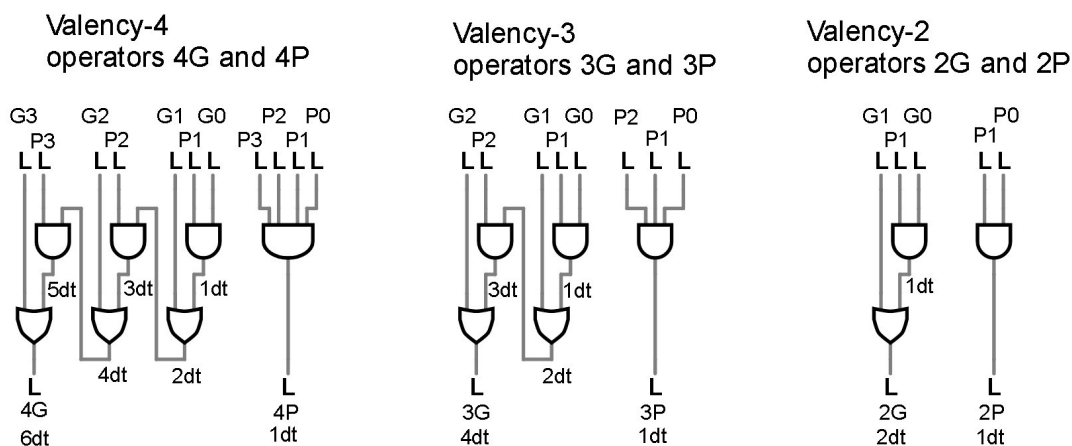


Рис. 2. Снимок моделей медленных логических операторов 4G,4P, 3G,3P и 2G,2P в симуляторе электроники Circuit Simulator.

[Включить Circuit Simulator с моделями медленных логических операторов.](#)

Времена выполнения медленных операторов:

2P - 1dt  
 2G - 2dt  
 3P - 1dt  
 3G - 4dt  
 4P - 1dt  
 4G - 6dt

Логические уравнения медленных операторов 4G,4P, 3G,3P и 2G,2P:

2P = P1 AND P0  
 2G = G1 OR (P1 AND G0)  
 3P = P2 AND P1 AND P0  
 3G = G2 OR (P2 AND (G1 OR (P1 AND P0)))  
 4P = P3 AND P2 AND P1 AND P0  
 4G = G3 OR (P3 AND (G2 OR (P2 AND (G1 OR (P1 AND G0))))))

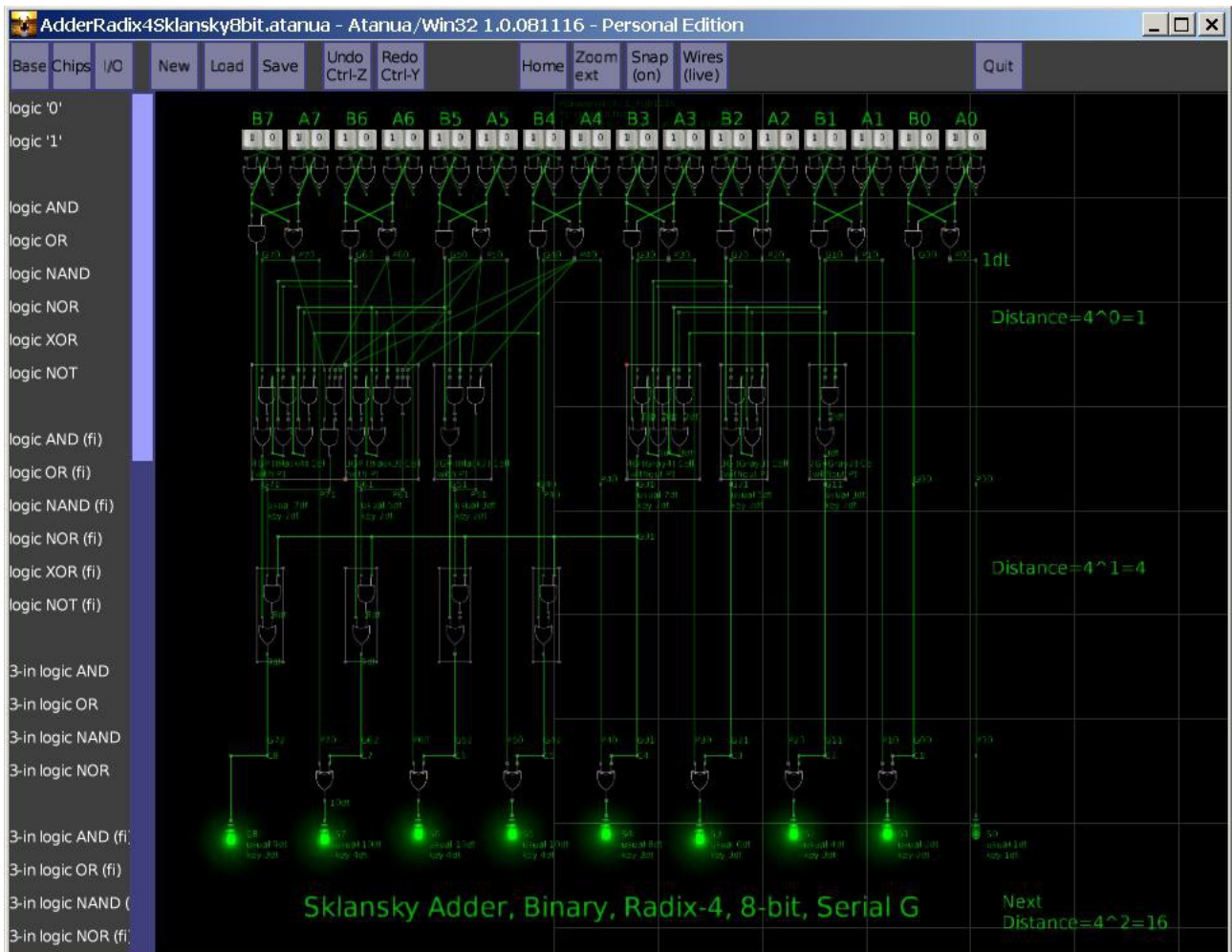


Рис.3. Снимок модели медленного двоичного сумматора Скланского, Radix-4, 8-ми битного (однобайтного) в логическом симуляторе [Atanua/Win32 1.0.081116 - Personal Edition](http://andserkul.narod.ru/AdderRadix4Sklansky8bit.atanua).

Код модели медленного двоичного сумматора Скланского, Radix-4, 8-ми битного (однобайтного) в логическом симуляторе Atanua/Win32:  
<http://andserkul.narod.ru/AdderRadix4Sklansky8bit.atanua>

Медленный двоичный сумматор Скланского, Radix-4, 8-ми битный (однобайтный), в виде системы логических уравнений:

```
'--- Step 0 -----
P00 = A0 XOR B0  '1dt
G00 = A0 AND B0  '1dt

P01 = A1 XOR B1  '1dt
G01 = A1 AND B1  '1dt

P02 = A2 XOR B2  '1dt
G02 = A2 AND B2  '1dt

P03 = A3 XOR B3  '1dt
G03 = A3 AND B3  '1dt

P04 = A4 XOR B4  '1dt
G04 = A4 AND B4  '1dt

P05 = A5 XOR B5  '1dt
```

```

G05 = A5 AND B5 '1dt
P06 = A6 XOR B6 '1dt
G06 = A6 AND B6 '1dt

P07 = A7 XOR B7 '1dt
G07 = A7 AND B7 '1dt

'--- Step 1 -----
G10 = G00 '1dt
G11 = G01 OR (P01 AND G00) '3dt
G12 = G02 OR (P02 AND (G01 OR (P01 AND G00))) '5dt
G13 = G03 OR (P03 AND (G02 OR (P02 AND (G01 OR (P01 AND G00)))) '7dt

P14 = P04 '1dt
G14 = G04 '1dt

P15 = P05 AND P04 '2dt
G15 = G05 OR (P05 AND G04) '3dt

P16 = P06 AND P05 AND P04 '2dt
G16 = G06 OR (P06 AND (G05 OR (P05 AND G04))) '5dt

P17 = P07 AND P06 AND P05 AND P04 '2dt
G17 = G07 OR (P07 AND (G06 OR (P06 AND (G05 OR (P05 AND G04)))) '7dt

'--- Step 2 -----
G20 = G10 '1dt, C1
G21 = G11 '3dt, C2
G22 = G12 '5dt, C3
G23 = G13 '7dt, C4
G24 = G14 OR (P14 AND G13) '9dt, C5
G25 = G15 OR (P15 AND G13) '9dt, C6
G26 = G16 OR (P16 AND G13) '9dt, C7
G27 = G17 OR (P17 AND G13) '9dt, C8

'--- Sum -----
S0 = P00 ' 1dt
S1 = P01 XOR G20 ' 2dt
S2 = P02 XOR G21 ' 4dt
S3 = P03 XOR G22 ' 6dt
S4 = P04 XOR G23 ' 8dt
S5 = P05 XOR G24 '10dt
S6 = P06 XOR G25 '10dt
S7 = P07 XOR G26 '10dt
S8 = G27 ' 9dt, S8=C8=Cout

```

Программа проверки системы логических уравнений медленного двоичного сумматора Склянского, Radix-4, 8-ми битного (однобайтного), на TurboBasic'e: <https://andserkul.narod.ru/R4SKL8.bas>

## **Быстрый (с параллельными операторами G) сумматор Склянского, Radix-4, 8-ми битный**

Для раскрытия скобок удобно формально заменить оператор OR на знак «+», а оператор AND на знак «\*», тогда раскрытие скобок в [алгебре логики](#) будет подобно раскрытию скобок в алгебре чисел:

```

2P = P1*P0
2G = G1 + P1*G0

3P = P2*P1*P0

```

$$3G = G2 + P2*(G1 + P1*G0)$$

$$4P = P3*P2*P1*P0$$

$$4G = G3 + P3*(G2 + P2*(G1 + P1*G0))$$

После раскрытия скобок последовательные операторы 3G и 4G становятся параллельными:

$$3G = G2 + P2*G1 + P2*P1*G0$$

$$4G = G3 + P3*G2 + P3*P2*G1 + P3*P2*P1*G0$$

Из полученных уравнений следует, что для логического вычисления параллельного оператора 3G понадобятся логические элементы 3-in OR, 2-in AND и 3-in AND, а для логического вычисления параллельного оператора 4G - 4-in OR, 2-in AND, 3-in AND и 4-in AND.

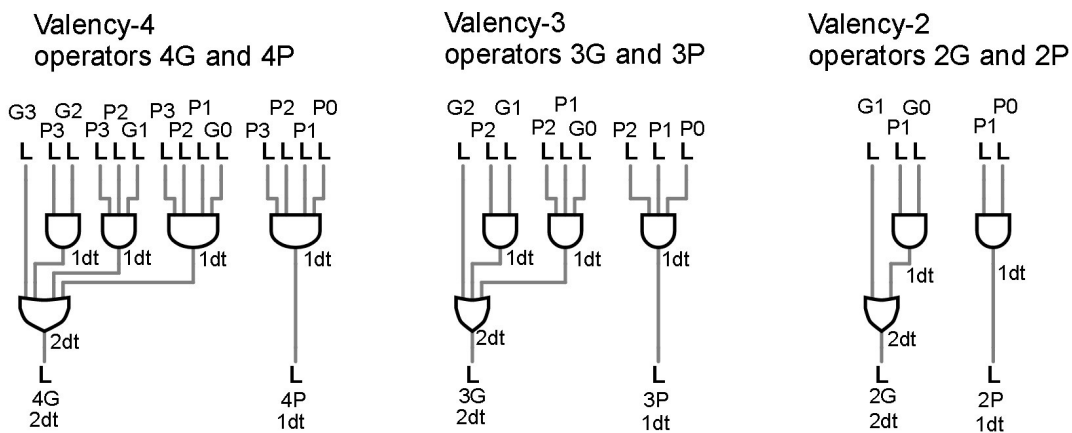


Рис. 4. Снимок моделей быстрых логических операторов 4G,4P, 3G,3P и 2G,2P в симуляторе электроники Circuit Simulator.

[Включить Circuit Simulator с моделями.](#)

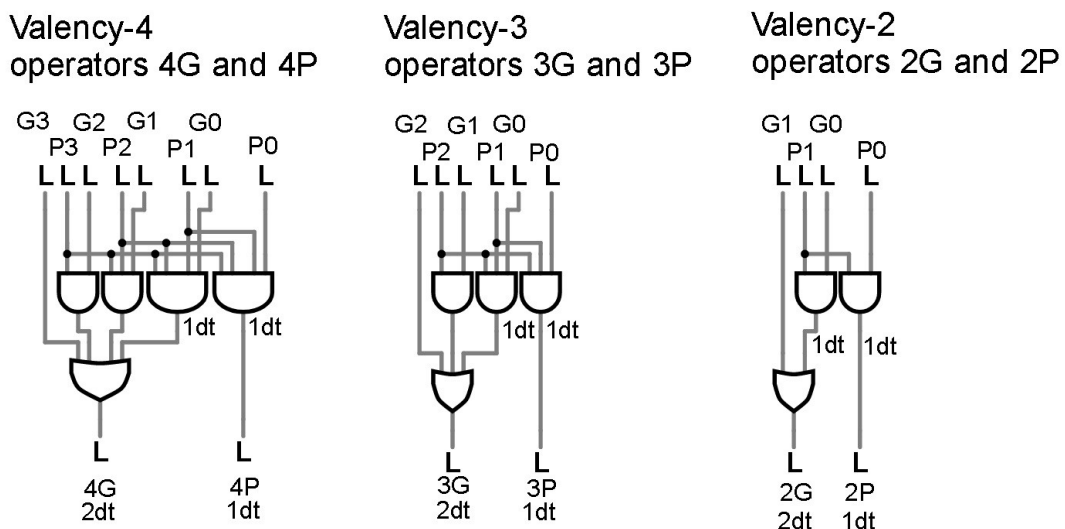


Рис. 5. Снимок упрощённых моделей быстрых логических операторов 4G,4P, 3G,3P и 2G,2P в симуляторе электроники Circuit Simulator.

Включить Circuit Simulator с моделями логических операторов.

Времена выполнения операторов:

- 2P – 1dt
- 2G – 2dt
- 3P – 1dt
- 3G – 2dt
- 4P – 1dt
- 4G – 2dt



Рис.6. Снимок модели быстрого двоичного сумматора Склянского, Radix-4, 8-ми битного, в логическом симуляторе Atanua/Win32 1.0.081116 - Personal Edition.

Код модели быстрого двоичного сумматора Склянского, Radix-4, 8-ми битного, в логическом симуляторе Atanua/Win32:

<https://andserkul.narod.ru/AdderRadix4Sklansky8bitF.atanua>

Быстрый двоичный сумматор Склянского, Radix-4, 8-ми битный, в виде системы логических уравнений:

```
'--- Step 0 -----
S00 = A0 XOR B0 '1dt
C00 = A0 AND B0 '1dt

S01 = A1 XOR B1 '1dt
C01 = A1 AND B1 '1dt

S02 = A2 XOR B2 '1dt
C02 = A2 AND B2 '1dt
```

```

S03 = A3 XOR B3 '1dt
C03 = A3 AND B3 '1dt

S04 = A4 XOR B4 '1dt
C04 = A4 AND B4 '1dt

S05 = A5 XOR B5 '1dt
C05 = A5 AND B5 '1dt

S06 = A6 XOR B6 '1dt
C06 = A6 AND B6 '1dt

S07 = A7 XOR B7 '1dt
C07 = A7 AND B7 '1dt

```

```

'--- Step 1 -----
G10 = C00 '1dt
G11 = C01 OR_ '3dt
      S01 AND C00

G12 = C02 OR_ '3dt
      S02 AND C01 OR_
      S02 AND S01 AND C00

G13 = C03 OR_ '3dt
      S03 AND C02 OR_
      S03 AND S02 AND C01 OR_
      S03 AND S02 AND S01 AND C00

P14 = S04 '1dt
G14 = C04 '1dt

P15 = S05 AND S04 '2dt
G15 = C05 OR_ '3dt
      S05 AND C04

P16 = S06 AND S05 AND S04 '2dt
G16 = C06 OR_ '3dt
      S06 AND C05 OR_
      S06 AND S05 AND C04

P17 = S07 AND S06 AND S05 AND S04 '2dt
G17 = C07 OR_ '3dt
      S07 AND C06 OR_
      S07 AND S06 AND C05 OR_
      S07 AND S06 AND S05 AND C04

```

```

'--- Step 2 -----
C20 = G10 '1dt, C0
C21 = G11 '3dt, C1
C22 = G12 '3dt, C2
C23 = G13 '3dt, C3
C24 = G14 OR (P14 AND G13) '5dt, C4
C25 = G15 OR (P15 AND G13) '5dt, C5
C26 = G16 OR (P16 AND G13) '5dt, C6
C27 = G17 OR (P17 AND G13) '5dt, C7

```

```

'--- Sum -----
S0 = S00 '1dt
S1 = S01 XOR C20 '2dt
S2 = S02 XOR C21 '4dt
S3 = S03 XOR C22 '4dt
S4 = S04 XOR C23 '4dt
S5 = S05 XOR C24 '6dt
S6 = S06 XOR C25 '6dt
S7 = S07 XOR C26 '6dt

```

$C7 = C27 \cdot 5dt, C7=Cout$

Время выполнения сложения в TTL — **6dt**, что на **2dt** быстрее, чем в сумматоре Склянского, Radix-2, 8-ми битном TTL (**8dt**), а в логике на ключах — KEY:**4dt**.

Программа проверки системы логических уравнений быстрого двоичного сумматора Склянского, Radix-4, 8-ми битного, на TurboBasic'e:

<https://andserkul.narod.ru/R4SKL8F.bas>

## **Быстродействие быстрых (с параллельными операторами G) сумматоров Склянского и Когге-Стоуна, на элементах типа TTL**

Количество разрядов	n=2	n=4	n=8	n=16	n=32	n=64
radix-2						
Количество шагов	s=1	s=2	s=3	s=4	s=5	s=6
Время сложения	4dt	6dt	8dt	10dt	12dt	14dt
radix-4						
Количество шагов		s=1	s=2	s=2	s=3	s=3
Время сложения		4dt	6dt	6dt	8dt	8dt
radix-8						
Количество шагов			s=1	s=2	s=2	s=2
Время сложения			4dt	6dt	6dt	6dt
radix-16						
Количество шагов				s=1	s=2	s=2
Время сложения				4dt	6dt	6dt
radix-32						
Количество шагов					s=1	s=2
Время сложения					4dt	6dt
radix-64						
Количество шагов						s=1
Время сложения						4dt

## **Быстродействие быстрых (с параллельными операторами G) сумматоров Склянского и Когге-Стоуна, с операторами P и G на CMOS ключах (KEY)**

Количество разрядов	n=2	n=4	n=8	n=16	n=32	n=64
radix-2						
Количество шагов	s=1	s=2	s=3	s=4	s=5	s=6
Время сложения	3dt	4dt	5dt	6dt	7dt	8dt
radix-4						
Количество шагов		s=1	s=2	s=2	s=3	s=3
Время сложения		3dt	4dt	4dt	5dt	5dt
radix-8						
Количество шагов			s=1	s=2	s=2	s=2
Время сложения			3dt	4dt	4dt	4dt
radix-16						
Количество шагов				s=1	s=2	s=2
Время сложения				3dt	4dt	4dt
radix-32						
Количество шагов					s=1	s=2
Время сложения					3dt	4dt
radix-64						
Количество шагов						s=1
Время сложения						3dt

## **Разница в схемотехнике**



Логические уравнения операторов вычисления P и G одинаковы и для оператора вычисления P и G на обычной физической реализации логики (TTL и CMOS) и для оператора вычисления P и G на физической реализации логики на CMOS-ключах, но при физической реализации логики на CMOS-ключах суммарное время задержки (gate delay, delay time,  $n*dt$ ) вычисляется иначе и значительно меньше, чем на обычной логике, т. е. быстродействие оператора вычисления P и G при физической реализации логики на CMOS-ключах значительно выше. Блоки в рамках при физической реализации логики на CMOS-ключах вычисляют значения P и G за время равное  $1*dt$ , где dt – время задержки в одном типовом физическом логическом элементе.

Так как параллельно префиксные сумматоры, в том числе и сумматор Склянского, Radix-4, строятся не последовательным соединением блоков с единицей переноса на входе, а целиком двухаргументными (двухоперандными), то в них исчезают понятия «полусумматор» и «полный сумматор», но сохраняются понятия «двухаргументный» и «трёхаргументный» (с единицей переноса на входе), причём «трёхаргументные» (с единицей переноса на входе) теоретически возможны, но практически в них нет почти никакой нужды.

Литература:

1. [Logical Effort of Higher Valency Adders. David Harris. Harvey Mudd College 301 E. Twelfth St. Claremont, CA 91711](#)
2. [Design Space Exploration for Power-Efficient Mixed-Radix Ling Adders. Chung-Kuan Cheng Computer Science and Engineering Depart. University of California, San Diego.](#)
3. [Сумматор Склянского, Radix-2, 4-х битный. Куликов А. С.](#)
4. [Сумматор Склянского, Radix-2, 8-ми битный. Куликов А. С.](#)
5. [Сумматор Склянского, Radix-2, 16-ти битный. Куликов А. С.](#)
6. [Сумматор Склянского, Radix-3, 4-х битный. Куликов А. С.](#)
7. [Сумматор Склянского, Radix-3, 8-ми битный. Куликов А. С.](#)
8. [Сумматор Склянского, Radix-3, 16-ти битный. Куликов А. С.](#)
9. [Сумматор Склянского, Radix-4, 4-х битный. Куликов А. С.](#)
10. [Сумматор Склянского, Radix-4, 16-ти битный. Куликов А. С.](#)
11. [Сумматор Склянского, Radix-4, 32-х битный. Куликов А. С.](#)
12. [Сумматор Склянского, Radix-8, 8-ми битный. Куликов А. С.](#)
13. [Сумматор Склянского, Radix-8, 16-ти битный. Куликов А. С.](#)
14. [Сумматор Склянского, Radix-8, 32-х битный, быстрый. Куликов А. С.](#)
15. [Сумматор Склянского, Radix-16, 16-ти битный. Куликов А. С.](#)

Приложение 1.

[TurboBasic 1.0](#)

Описание TurboBasic'a:

<https://andserkul.narod.ru/TBASIC.txt>

Куликов А.С., Россия-Русь, Москва, Царицыно, версия 2024.11.28.