

Потоковое шифрование
Яковлев Вадим, гр. 051003
Вариант 5

Проверка генератора LFSR:

Полином: $x^{27}+x^8+x^7+x+1=0$

Начальное состояние регистра: 27 единиц.

Исходные данные для шифрования (байтовое представление):

11010000 10110000 11010000 10110001 11010000 10110011

Ожидаемый ключ (столбец регистра с номером 27):

11111111 11111111 11111111 11101010 10000000 01010101

		27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1		
Шаги / Состояние регистра	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0		
	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	
	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	
	4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	
	5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	
	6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	
	7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	
	8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	
	9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	
	10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	
	11	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	
	12	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	
	13	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	
	14	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	
	15	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1
	16	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0
	17	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1
	18	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0
	19	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1
	20	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0
	21	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1
	22	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1
	23	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1
	24	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1
	25	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1
	26	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1
	27	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1
	28	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0
	29	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1
	30	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1
	31	1	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0
	32	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0
	33	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1
	34	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1
	35	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0
	36	0	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1
	37	0	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1
	38	0	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0
	39	0	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	0
	40	0	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	1	1
	41	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	1	1	1
	42	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	1	1	1	1
	43	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	1	1	1	1	0
	44	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	1	1	1	1	0	1
	45	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	1	1	1	1	0	1	0
	46	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	1	1	1	1	0	1	0	1
	47	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	1	1	1	0	1	0	1	0	1
	48	1	1	1	1	1	1	1	0	1	1	0	0	1	1	0	1	1	0	0	1	1	1	0	1	0	1	0	1	0

Полученный ключ:

```
11111111 11111111 11111111 11101010
10000000 01010101
```

Проверка шифрования:

Исходные данные (в байтах):

11010000 10110000 11010000 10110001 11010000 10110011

Ключ (в байтах):

11111111 11111111 11111111 11101010 10000000 01010101

11010000 10110000 11010000 10110001 11010000 10110011

xor

11111111 11111111 11111111 11101010 10000000 01010101

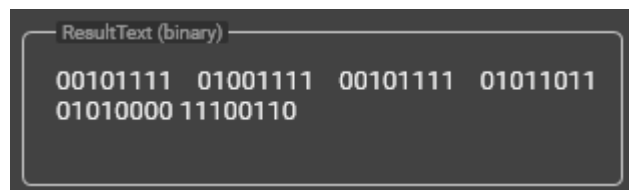
=

00101111 01001111 00101111 01011011 01010000 11100110

Ожидаемый результат (в байтах):

00101111 01001111 00101111 01011011 01010000 11100110

Полученный результат:



The screenshot shows a terminal window with a dark background. At the top, there is a label "ResultText (binary)". Below it, the result of the XOR operation is displayed in two lines of white text: "00101111 01001111 00101111 01011011" on the first line and "01010000 11100110" on the second line.