# Обращение криптографической функции A5/1 на платформе GPU с применением альтернативных схем вычисления сдвиговых регистров*

В.Г. Булавинцев, А.А Семенов

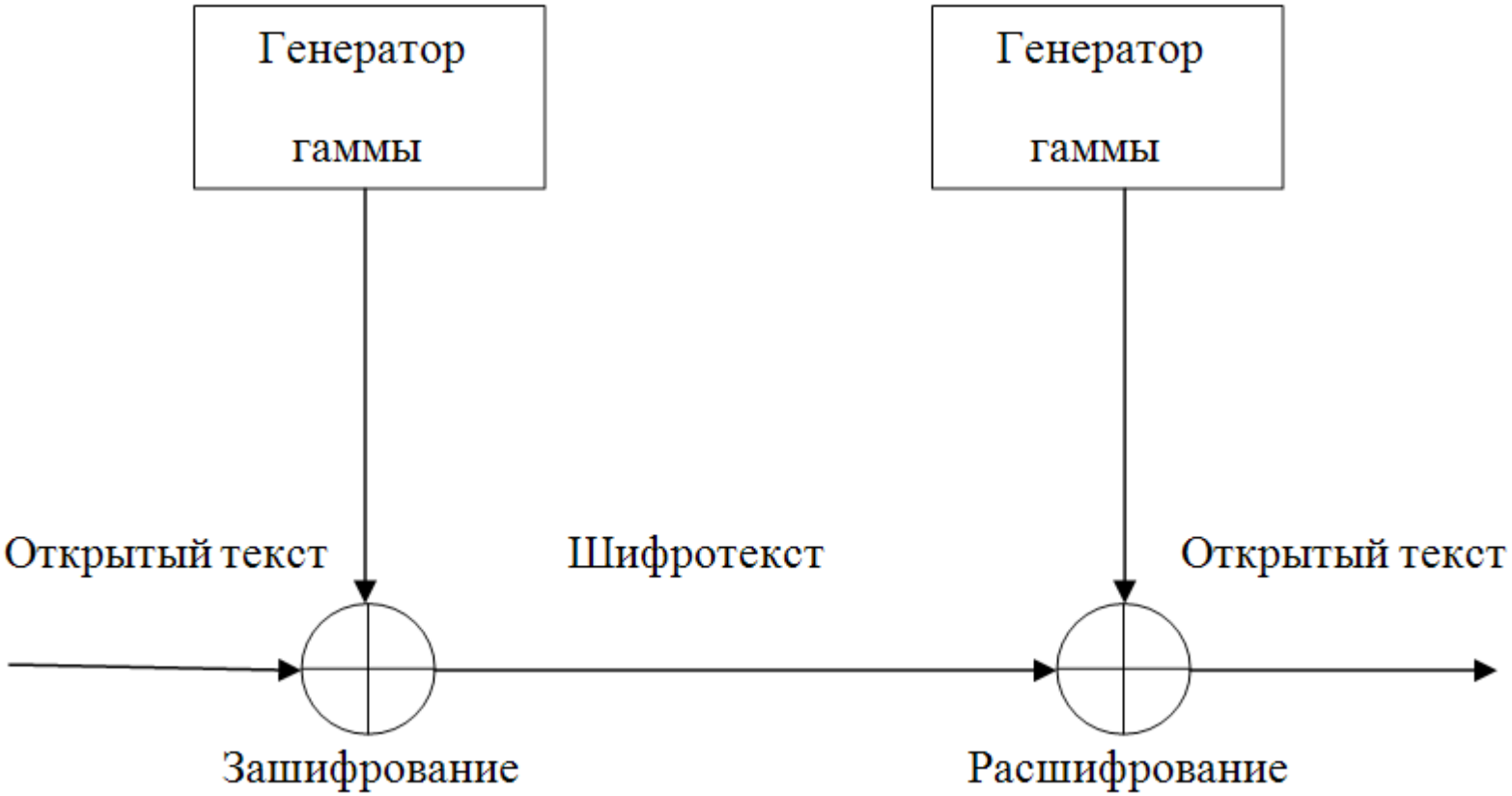
ИДСТУ СО РАН
2016

*Работа была частично поддержана РФФИ (№ 14-07-00403-а, 15-07-07891-а и 16-07-00155-а)

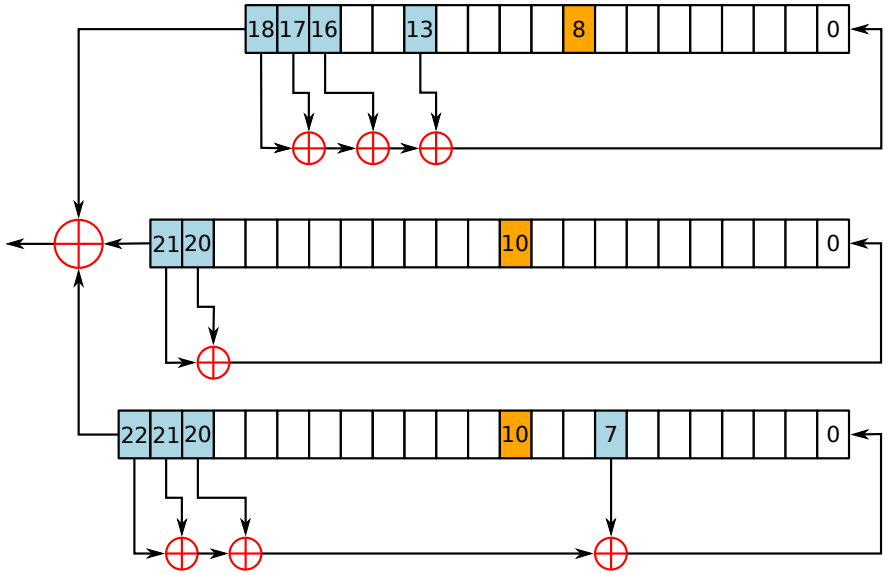
# Стандарт GSM и генератор A5/1

- Стандарт цифровой мобильной связи GSM принят в 1991 году.
- Основной элемент криптографической защиты передачи данных в GSM — генератор потокового шифра A5/1.
- Длина ключа A5/1 — **64 бита**.
- Передача шаблонных сообщений инициализации GSM в зашифрованном виде, является уязвимостью протокола и позволяет применить к A5/1 «атаку в условиях открытого текста».

# Потоковая криптосиситема



# Генератор потокового шифра A5/1




$$x^{19}+x^{18}+x^{17}+x^{14}+1;$$
$$x^{22}+x^{21}+1;$$
$$x^{23}+x^{22}+x^{21}+x^{8}+1;$$
$$b_j = maj(b_1, b_2, b_3);$$
$$j - номер\ РСЛОС.$$

- 3 регистра сдвига с линейной обратной связью (РСЛОС): 19, 22, 23 разряда (в сумме 64). Выходы смешиваются по XOR.
- **Условное тактирование** РСЛОСов по «функции большинства» взятой от серединных бит:

$$maj(A,B,C) = (A \cap B) \cup (A \cap C) \cup (B \cap C).$$

# Атаки на A5/1

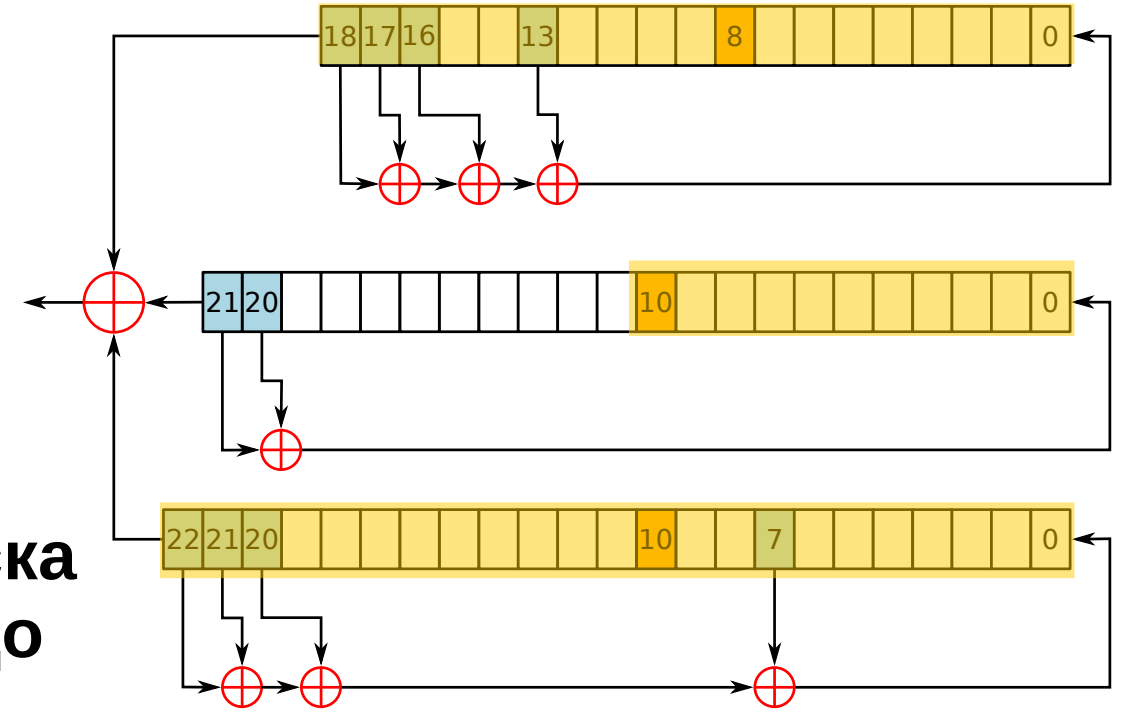
- «Атака Андерсона» предложена в 1994 году. Позволяет уменьшить пространство поиска при атаке методом прямого перебора.
- Атаки пространственно-временного компромисса: Biryukov, Shamir, Wagner, 2000; Barkhan, Biham, Keller, 2007.
- Атака с применением SAT-подхода: Посыпкин, Заикин, Беспалов, Семенов, 2009 (2008).
- Открыто доступные rainbow-таблицы представлены «A5/1 cracking project» в 2009.

# Ускорение атаки на A5/1

- «Атака Андерсона» позволяет сократить пространство поиска относительно метода прямого перебора в $2^{11}$ раз, до $2^{53}$.
- Скорость атаки напрямую зависит от эффективности программной реализации алгоритма генератора A5/1.
- Мы рассматриваем 2 быстрых реализации генератора A5/1:
  - с применением техники «bitslice»;
  - с использованием предвычисления последовательностей РСЛОСов.

# «Атака Андерсона», шаг 1

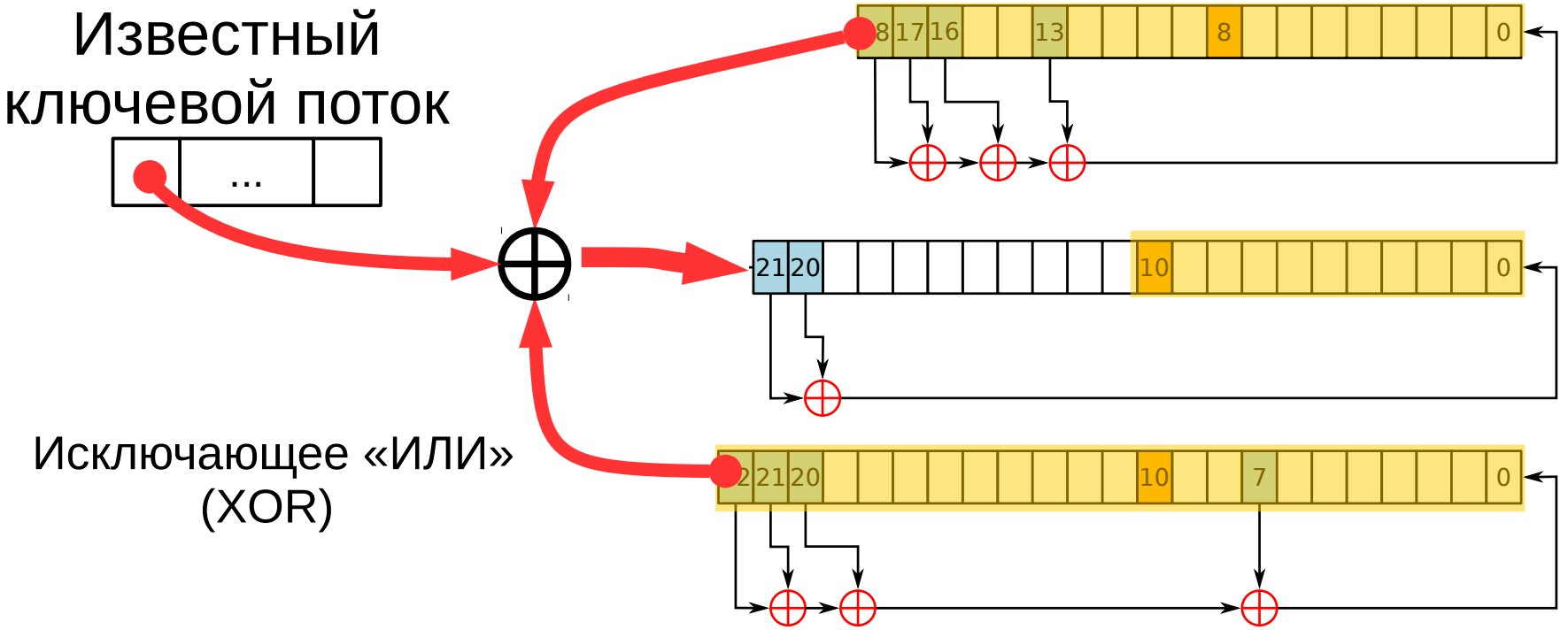
- **Угадать** заполнение 1 и 3 регистров полностью и заполнение 2-го регистра справа от серединного бита.




Угадываем значения 53 разрядов из 64. **Пространство поиска сокращается с $2^{64}$ до $2^{53}$.**

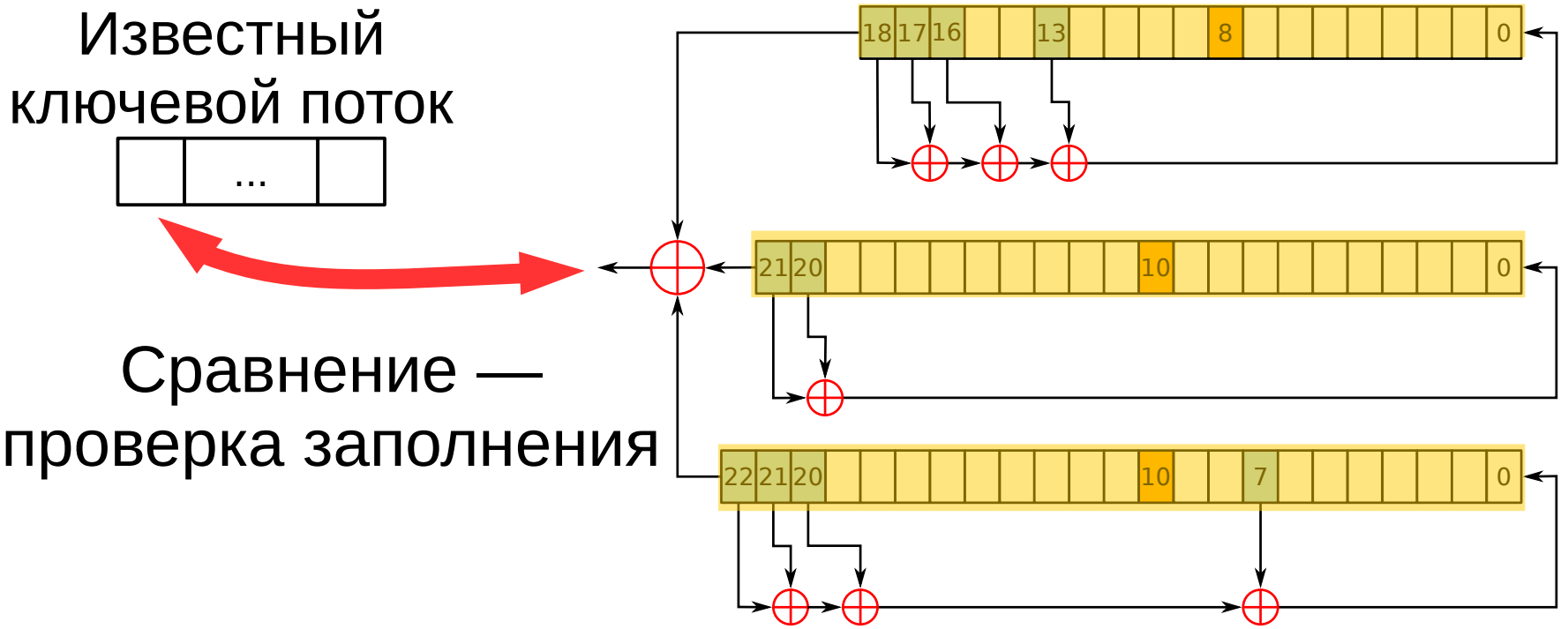
# «Атака Андерсона», шаг 2

- По известному фрагменту ключевого потока, тактируя генератор, **вычислить** заполнение 2-го регистра слева от серединного бита.



# «Атака Андерсона», шаг 3

- **Проверить** полученное таким образом состояние генератора тактируя его и сравнивая с ключевым потоком



# Техника «bitslice» - «SIMD within a register»

- Использование **побитовых** логических инструкций (И, ИЛИ, НЕ и др.) позволяет одновременно вычислять столько значений булевой функции, сколько вмещает один регистр процессора (т. е. 32 и более).
- Чтобы это было возможно, требуется построить «схемное» представление алгоритма, использующее только побитовые инструкции, и не использующее оператор условного перехода.
- Такая схема часто используется для ускорения блочных шифров (DES и т.д.)

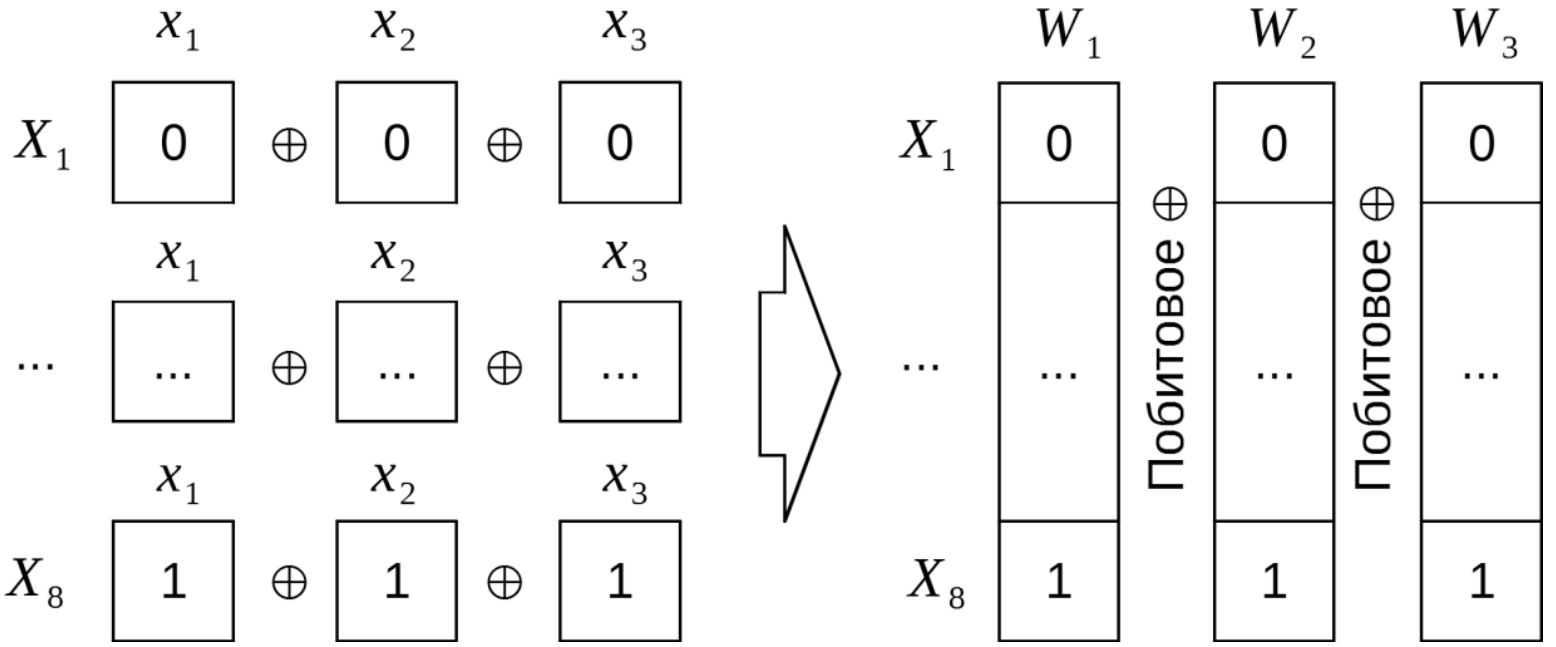
# Техника «bitslice» - «SIMD within a register»

$$f_{\oplus}:\{0,1\}^3=\{0,1\};$$
$$f_{\oplus}(x_1, x_2, x_3)=x_1 \oplus x_2 \oplus x_3.$$

На всех 8 входах:
16 инструкций АЛУ

$$W_1=(0,0,0,0,1,1,1,1);$$
$$W_2=(0,0,1,1,0,0,1,1);$$
$$W_3=(0,1,0,1,0,1,0,1);$$
$$f'_{\oplus}(W_1, W_2, W_3)=W_1 \oplus W_2 \oplus W_3.$$

2 инструкции АЛУ .



# «Bitslice»-реализация A5/1. Сдвиги РСЛОСов.

- Сдвиги РСЛОСов $R_1$, $R_2$, $R_3$ осуществляются последовательным переприсвоением значений переменных:

$$W_n \in \{0,1\}^k, k - \textit{разрядность платформы};$$
$$R1: W_{1,}..., W_{19};$$
$$W'_1 = W_{19} \oplus W_{18} \oplus W_{17} \oplus W_{14};$$
$$W'_n = W_{n-1}, n \in \{2,..., 19\};$$

# «Bitslice»-реализация A5/1. Вычисление «majority».

- Вычисление функции «majority» проводится аналогично «наивной» реализации:

$$W_{maj} = maj(W_9, W_{33}, W_{54}) =$$
$$= (W_9 \cap W_{33}) \cup (W_9 \cap W_{54}) \cup (W_{33} \cap W_{54});$$
$$F_{R_1} = W_9 \oplus W_{maj};$$
$$F_{R_2} = W_{33} \oplus W_{maj};$$
$$F_{R_3} = W_{54} \oplus W_{maj}.$$

# «Bitslice»-реализация A5/1. Условное тактирование.

- Условное тактирование РСЛОСов в рамках схемы «bitslice» требует возможности независимого условного присвоения значений переменных **без применения оператора условного перехода**.
- Применение функции арности 3 «выбор бита» («bitselect») позволяет решить эту проблему:

$$x, y, z \in \{0, 1\};$$

$$BS(x, y, z) = \begin{cases} y, x = 1 \\ z, x = 0 \end{cases};$$

$$W'_1 = BS(F_{R_1}, (W_{19} \oplus W_{18} \oplus W_{17} \oplus W_{14}), W_1);$$

$$W'_n = BS(F_{R_1}, W_{n-1}, W_n), n \in \{2, \ldots, 19\}.$$

# «Bitslice»-реализация A5/1. Пример кода тактирования РСЛОС.

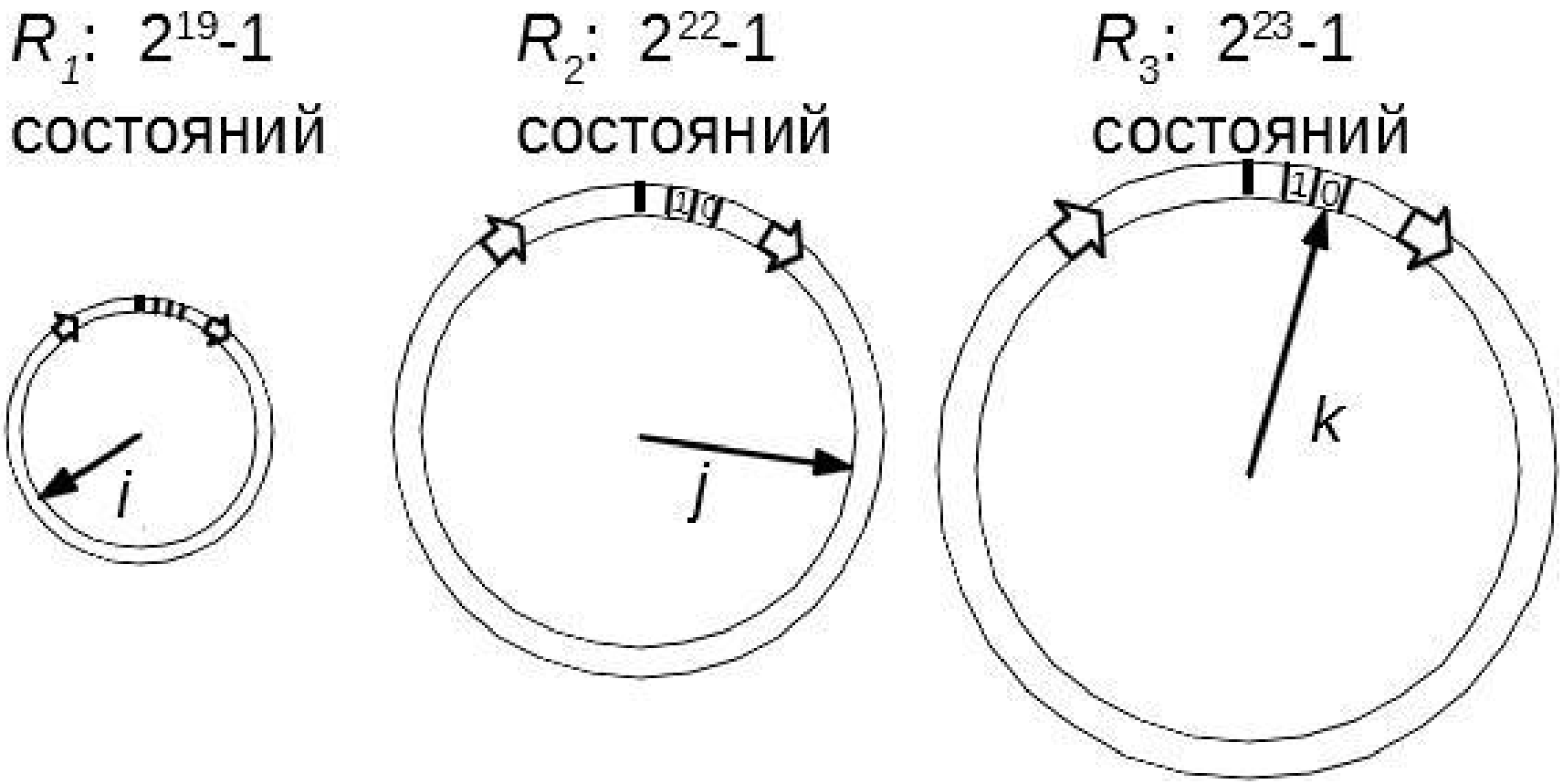
```
// Разряды РСЛОСов:
uint32 x01, x02, …, x19; // R1
uint32 x20, x41, …, x41; // R2
uint32 x42, x43, …, x64; // R3
...
// Вычисление флага сдвига РСЛОС R_1:
F1 = x09 ^ maj (x09, x33, x54);

// Условный сдвиг разрядов РСЛОСа
xn  = x19 ^ x18 ^ x17 ^ x14;
x19 = bs (F1, x18, x19);
x18 = bs (F1, x17, x18);
...
x02 = bs (F1, x01, x02);
x01 = bs (F1, xn,  x01);
```

# Реализация A5/1 с предвычислением РСЛОСов

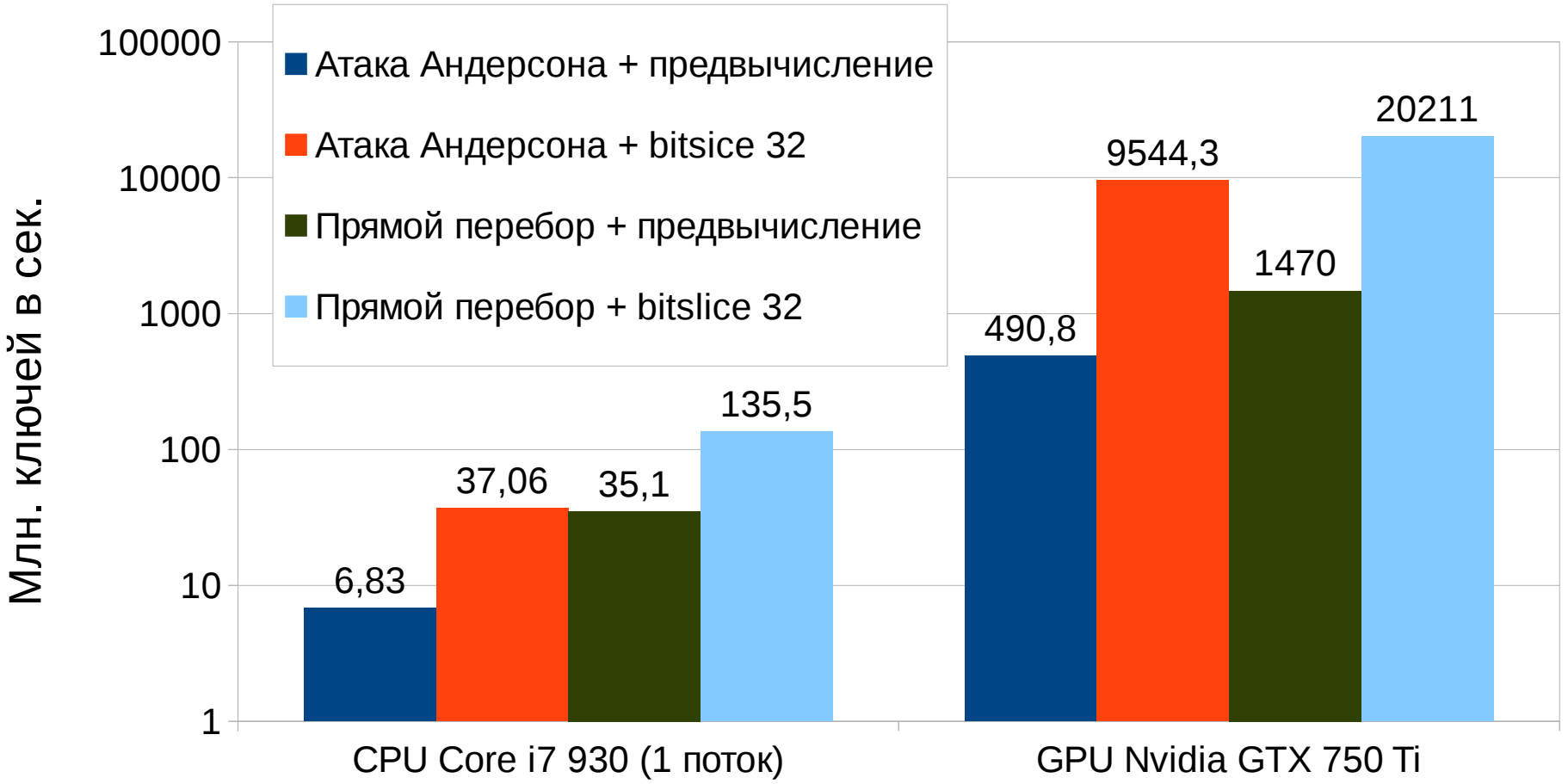
- Впервые техника предложена в 2000 году в работе Biryukov A., Shamir A., Wagner D. Real «Time Cryptanalysis of A5/1 on a PC»
- Поскольку РСЛОС в A5/1 короткие, не зависят друг от друга и порождают последовательности максимальной длины, можно **однократно сгенерировать** эти последовательности и хранить в памяти в виде круговых битовых массивов.
- Теперь для получения состояния ***i***-го разряда РСЛОСа на ***n***-ом такте достаточно извлечь ***i+n*** элемент из соответствующего битового массива.
- Условное тактирование обеспечивается индивидуальными счетчиками текущего такта для каждого РСЛОСа.

# Циклические массивы с предвычисленными состояниями РСЛОСов




Суммарный размер памяти, занимаемый массивами - около 1,5 Мб

# Экспериментальные результаты: скорость перебора на GPU и CPU




**Пространство поиска «атаки Андерсона» в 2048 раз меньше, чем у прямого перебора!**

# Заключение

- Применение «атаки Андерсона», в сочетании с техникой bitslice позволяет достичь скорости перебора в 1470 * 10^6 ключей в секунду. Пространство перебора при этом составит 2^53 (~10^16).
- На маломощном потребительском GPU GTX 750 Ti криптоанализ A5/1 займет 270 часов.
  На современном HPC-кластере — минуты.
- В отличие от криптоанализа с применением Rainbow-таблиц (вероятность успеха 88%), применение «атаки Андерсона» гарантирует успех в 100% случаев.

Спасибо за внимание!