Об исследовании возможностей построения эффективных реализаций одного перспективного LSX-шифра

Смышляев С.В.
Алексеев Е.К., Попов В.О., Прохоров А.С., Сонина Л.А.

Введение

В предлагаемом коротком сообщении рассматривается перспективный алгоритм блокового шифрования, предложенный в докладе на РусКрипто'2013 и построенный по LSX-схеме.

Далее указанный алгоритм будет обозначать K-128.

Размеры блока и ключа - 128 и 256 бит соответственно.

В предлагаемом сообщении представлены предварительные результаты теоретических и экспериментальных исследований особенностей конструкции шифра K-128 с точки зрения возможности построения эффективных реализаций на базе существующего поколения CPU с поддержкой SIMD-расширений и на GPU.

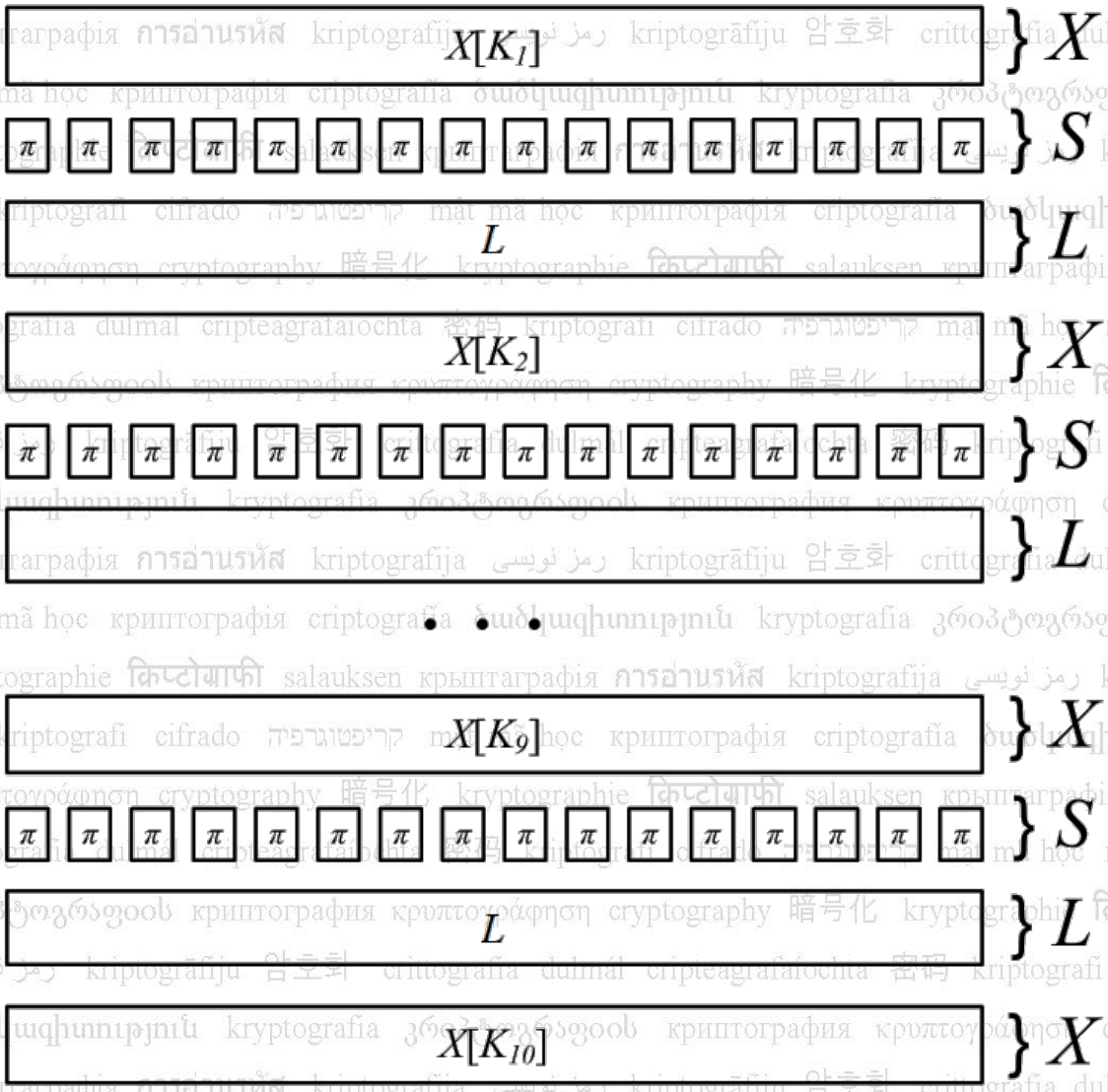
Введение

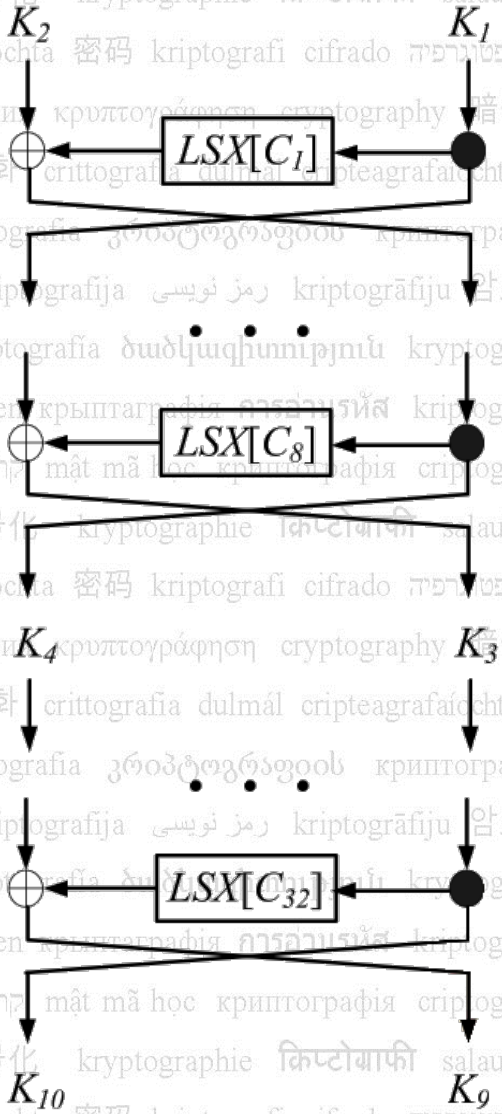
В предлагаемом коротком сообщении рассматривается перспективный алгоритм блокового шифрования, предложенный в докладе на РусКрипто'2013 и построенный по LSX-схеме.

Далее указанный алгоритм будет обозначать K-128.

Размеры блока и ключа - 128 и 256 бит соответственно.

В предлагаемом сообщении представлены предварительные результаты теоретических и экспериментальных исследований особенностей конструкции шифра K-128 с точки зрения возможности построения эффективных реализаций на базе существующего поколения CPU с поддержкой SIMD-расширений и на GPU.





Основное преобразование имеет вид $L(S(X[K](in)))$, где

- $X[K]$ — сложение по модулю 2 с набором K ($X[K](z) = z \oplus K$),
- S — побайтовая подстановка в соответствии с биекцией $\pi : V_8 \to V_8$: $S(x_{15}x_{14}\ldots x_1x_0) = \pi(x_{15})\pi(x_{14})\ldots\pi(x_1)\pi(x_0)$,
- L — умножение на 16×16 матрицу A с элементами из $GF(2^8)$.

Зашифрование: $E_{K_1,\ldots,K_{10}} = X[K_{10}]LSX[K_9]\ldots LSX[K_2]LSX[K_1]$.
Расшифрование: $D_{K_1,\ldots,K_{10}} = X[K_1]S^{-1}L^{-1}\ldots X[K_9]S^{-1}L^{-1}X[K_{10}]$.
Раундовые ключи $K_i \in V_{128}$ вычисляются по ключу $K \in V_{256}$ по следующей схеме:

$$K \in V_{256}, (K_1, K_2) = K,$$

$$(K_{2i+1}, K_{2i+2}) = F[C_{8(i-1)+8}]\ldots F[C_{8(i-1)+1}](K_{2i-1}, K_{2i}), i = 1, \ldots, 4,$$

$$F[C] : V_{128}^2 \to V_{128}^2, F[C](a_1, a_0) = (LSX[C](a_1) \oplus a_0, a_1),$$

$$C_i \in V_{128}, C_i = L([i]_2), i = 1, \ldots, 32.$$

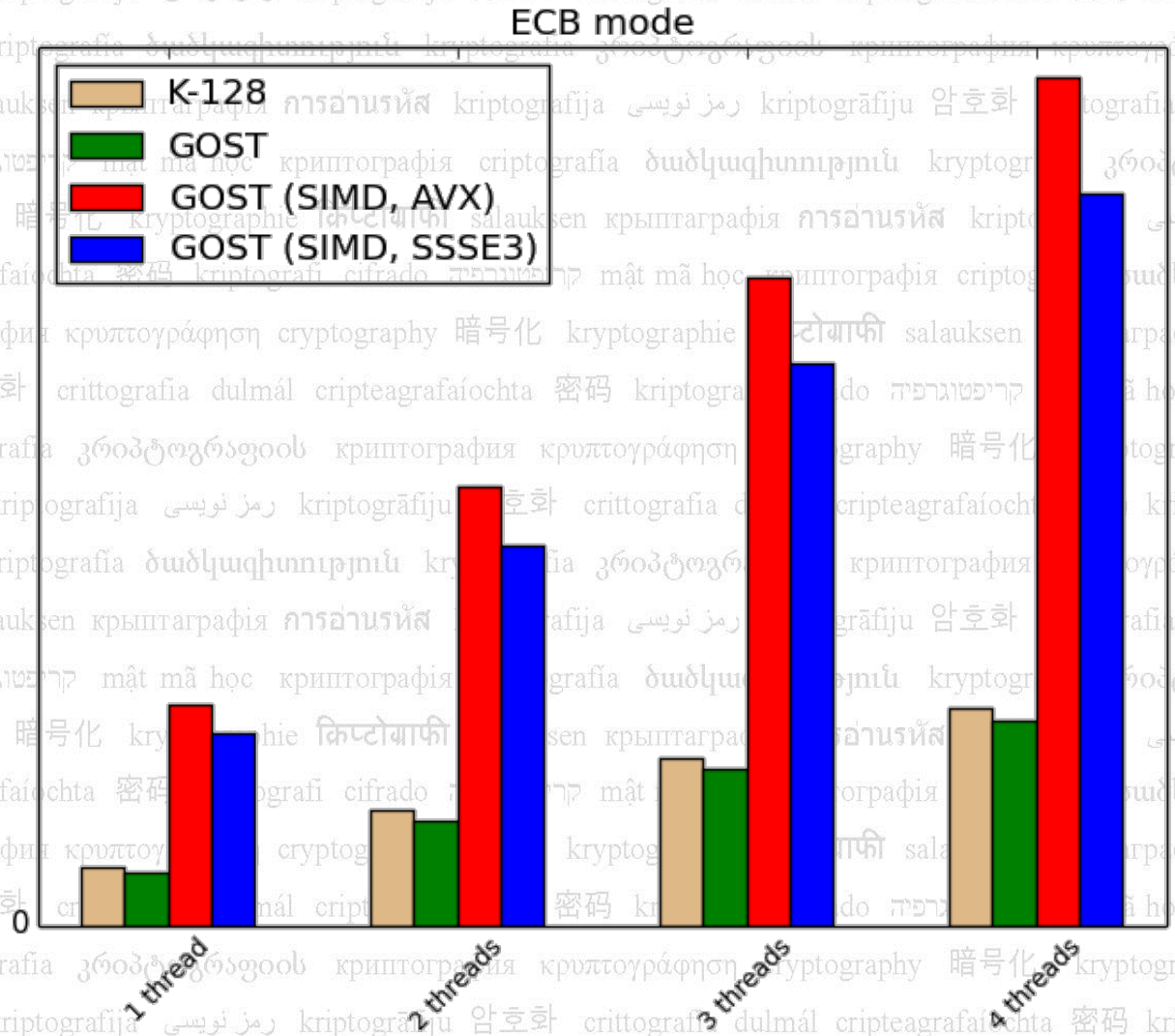
КРИПТО ПРО

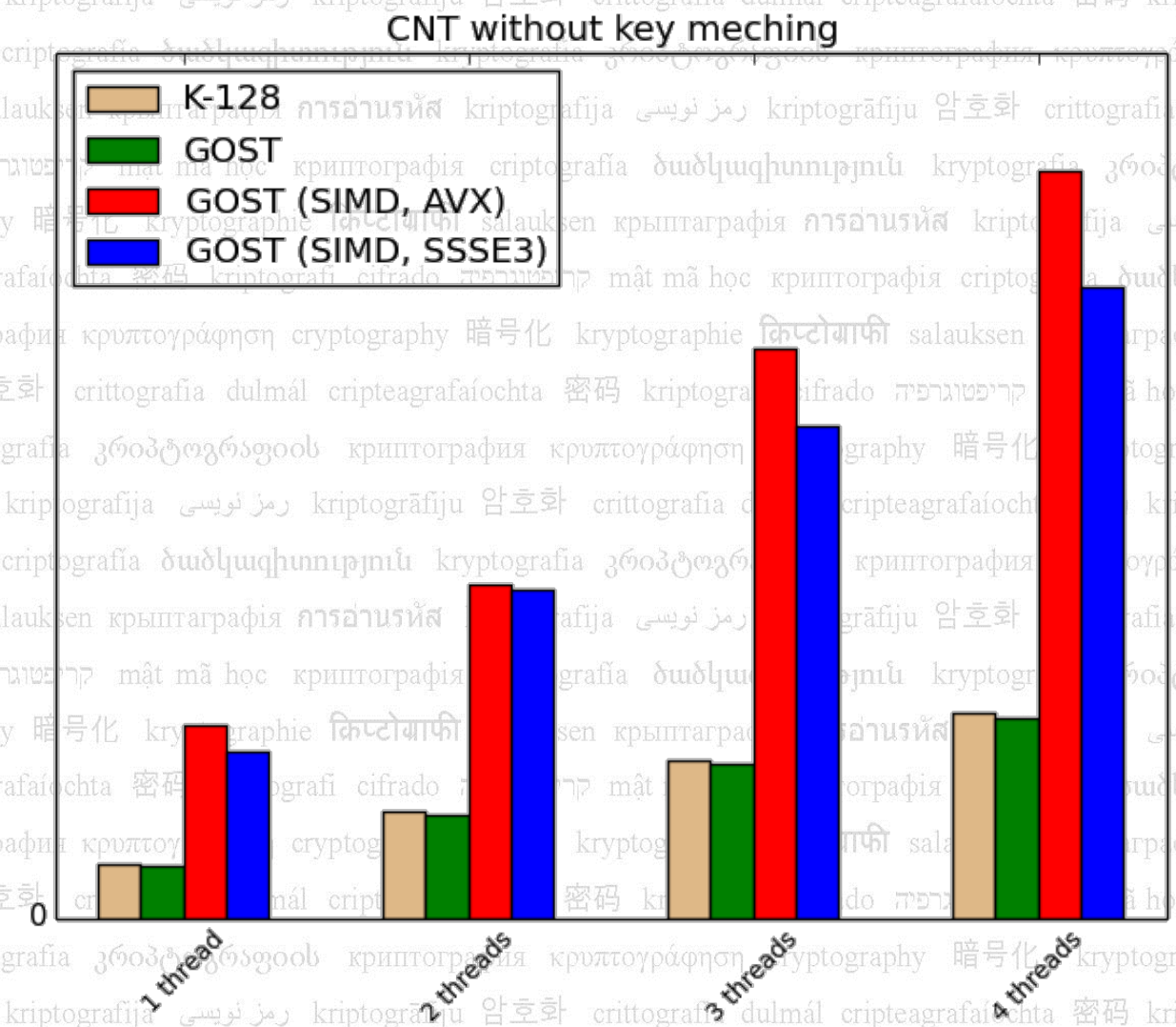
Сравниваем: K-128 и ГОСТ 28147-89.

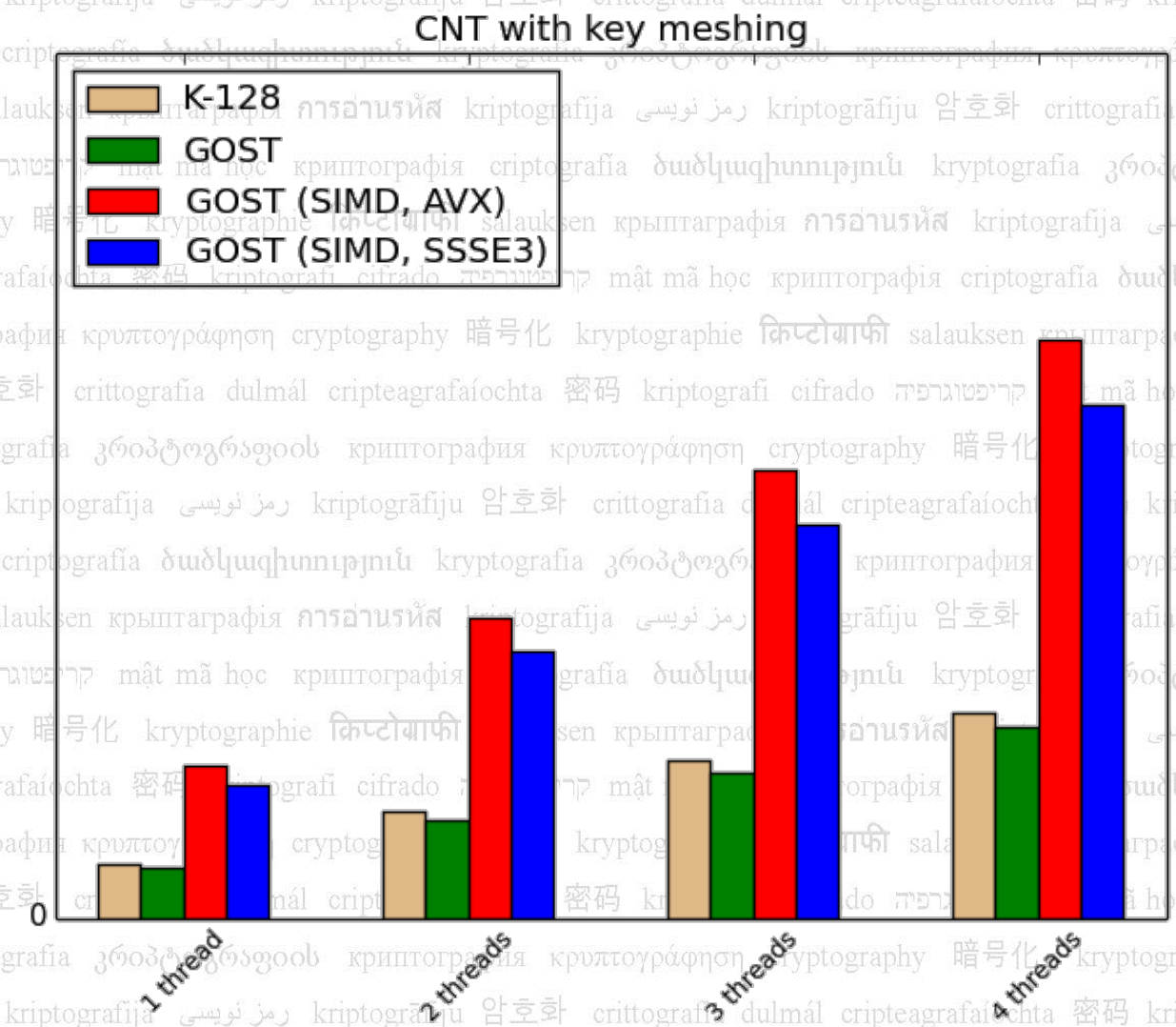
ГОСТ 28147-89: 32 раунда, сеть Фейстеля, простое ключевое расписание, блок 64 бита, наложение ключа по модулю 2^{32}, нелинейное преобразование осуществляется над полубайтами, линейное преобразование — сдвиг на 11.

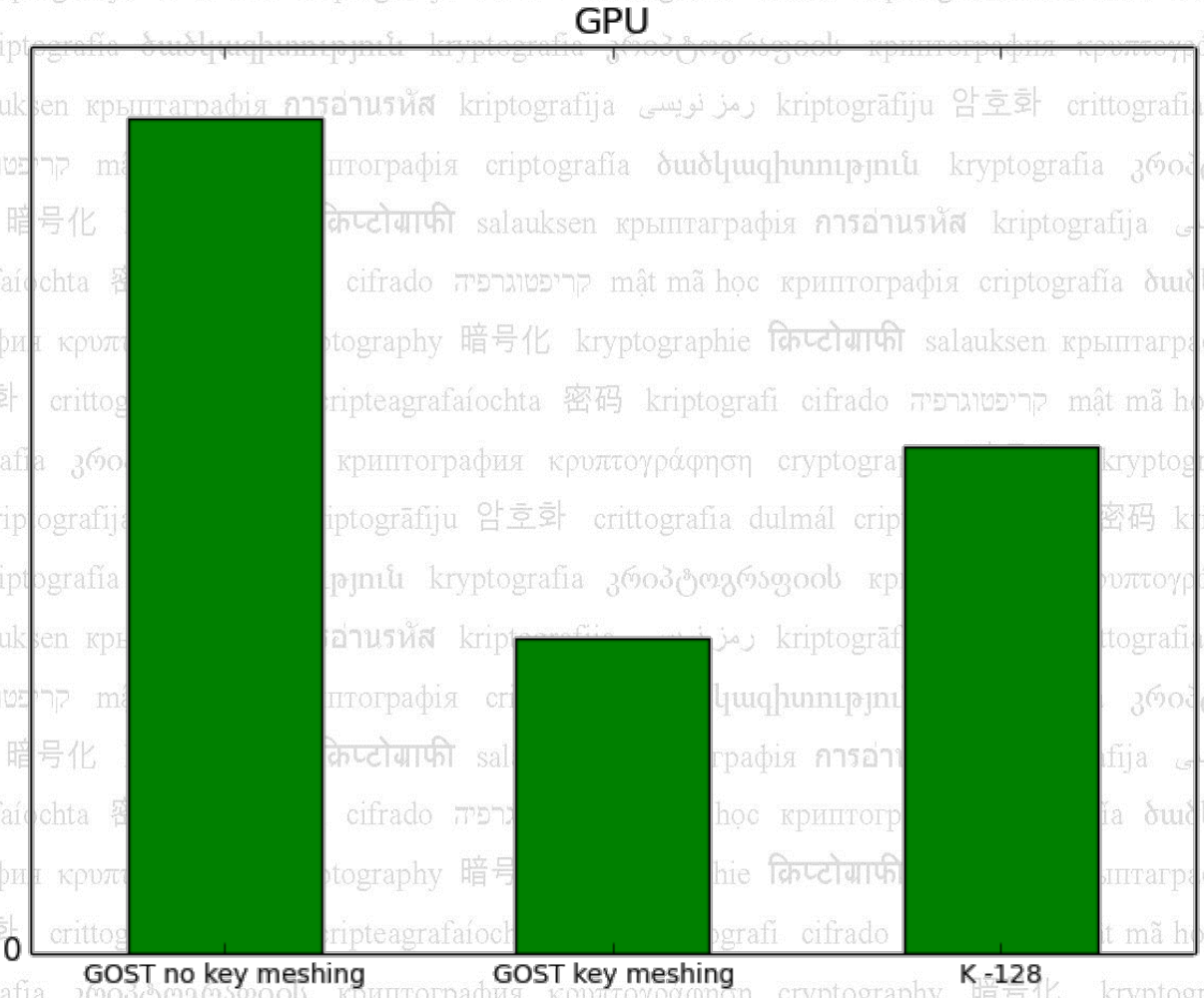
K-128: 9.5 раундов, LSX, ключевое расписание на основе LSX-преобразований, наложение ключа по модулю 2, блок 128 битов, нелинейное преобразование осуществляется над байтами, линейное преобразование — умножение на матрицу 16×16.

Неуместность сравнения с AES: с 2010 года все процессоры Intel Core^{TM} (32nm Westmere) содержат инструкции, специально созданные для поддержки шифрования по AES. Преимущество аппаратной поддержки.









- K-128 позволяет использовать крайне ограниченную часть возможностей, предоставляемых SIMD-расширением CPU. При этом их использование позволяет строить реализации, эффективность которых превышает эффективность реализации ГОСТ 28147-89 без использования SIMD-инструкций.
- Эффективное применение всего спектра возможностей, предоставляемых SIMD-расширением CPU, при реализации K-128 представляется невозможным. В связи с этим эффективность упомянутой выше реализации K-128 значительно ниже эффективности реализации ГОСТ 28147-89 с использованием SIMD-инструкций.
- K-128 допускает построение на GPU реализации более эффективных, чем GPU-реализации ГОСТ 28147-89. В основном выигрыш получается за счет возможности обработки большого объема данных без смены ключа.

- K-128 позволяет использовать крайне ограниченную часть возможностей, предоставляемых SIMD-расширением CPU. При этом их использование позволяет строить реализации, эффективность которых превышает эффективность реализации ГОСТ 28147-89 без использования SIMD-инструкций.
- Эффективное применение всего спектра возможностей, предоставляемых SIMD-расширением CPU, при реализации K-128 представляется невозможным. В связи с этим эффективность упомянутой выше реализации K-128 значительно ниже эффективности реализации ГОСТ 28147-89 с использованием SIMD-инструкций.
- K-128 допускает построение на GPU реализации более эффективных, чем GPU-реализации ГОСТ 28147-89. В основном выигрыш получается за счет возможности обработки большого объема данных без смены ключа.

- K-128 позволяет использовать крайне ограниченную часть возможностей, предоставляемых SIMD-расширением CPU. При этом их использование позволяет строить реализации, эффективность которых превышает эффективность реализации ГОСТ 28147-89 без использования SIMD-инструкций.
- Эффективное применение всего спектра возможностей, предоставляемых SIMD-расширением CPU, при реализации K-128 представляется невозможным. В связи с этим эффективность упомянутой выше реализации K-128 значительно ниже эффективности реализации ГОСТ 28147-89 с использованием SIMD-инструкций.
- K-128 допускает построение на GPU реализации более эффективных, чем GPU-реализации ГОСТ 28147-89. В основном выигрыш получается за счет возможности обработки большого объема данных без смены ключа.

КРИПТО ПРО

- Влияние компонент схемы на возможность построения эффективных SIMD-реализаций для ЦП.

КРИПТО ПРО

- Влияние компонент схемы на возможность построения эффективных SIMD-реализаций для ЦП.
- Влияние компонент схемы на возможность построения эффективных SIMD-реализаций для графических процессоров.
- Подтверждение выводов экспериментальными исследованиями различных классов модельных модификаций схемы.

КРИПТО ПРО

- Влияние компонент схемы на возможность построения эффективных SIMD-реализаций для ЦП.
- Влияние компонент схемы на возможность построения эффективных SIMD-реализаций для графических процессоров.
- Подтверждение выводов экспериментальными исследованиями различных классов модельных модификаций схемы.

КРИПТО ПРО

Спасибо за внимание!