

УДК xxx.yyy

*С. В. Дорохин^{1,2}, С. С. Качков^{1,3}, А. А. Сидоренко^{1,3}*¹Московский физико-технический институт²АО "ПКК Миландр"³АО "Интел"

Реализация шифра ГОСТ Р 34.12 2015 с использованием SIMD инструкций

В данной работе обсуждаются возможности увеличения скорости работы блочного шифра ГОСТ Р 34.12 2015, известного как "Кузнечик". В отличие от шифра AES, "Кузнечик" не имеет аппаратной поддержки на процессорах Intel и AMD, поэтому вопрос о максимально возможной скорости за- и расшифрования стоит особенно остро. В первой части работы кратко описано устройство шифра и рассмотрены известные на текущий момент методы ускорения, связанные с использованием таблиц предвычислений. Далее описываются опубликованные в свободном доступе варианты реализации, основанные на наборах векторных инструкций SSE и AVX. В следующей части мы представляем наши модификации упомянутых реализаций с использованием набора инструкций AVX2 для некоторых режимов работы шифра. В заключение приводится сравнительная характеристика скорости работы шифра в различных реализациях.

Ключевые слова: SIMD инструкции, Кузнечик, SSE, AVX, ГОСТ Р 34.12 2015, программная реализация.

*S. V. Dorokhin^{1,2}, S. S. Kachkov^{1,3}, A. A. Sidorenko^{1,3}*¹Moscow Institute of Physics and Technology²JSC "Milandr"³JSC "Intel"

Implementation of GOST R 34.12 2015 cipher using SIMD instructions

This article is dedicated to optimized implementations of GOST R 34.12 2015 block cipher. Compared with AES cipher, GOST cipher is not supported on hardware level by any known processors. That is why the task of exploring its maximum encryption/decryption speed is a crucial one. In the first section the cipher's algorithm and a brief description of relatively trivial LUT (lookup tables) optimizations are given. The next paragraph consists of a concise overview of currently available implementations which use vector command sets SSE and AVX. Finally, our AVX-based modifications are presented followed by a comparative analysis of open-source implementations.

Key words: SIMD instructions, SSE, AVX, GOST R 34.12 2015, block cipher, software implementation.

1. Введение

Для полноты изложения приведём краткое описание алгоритма, более подробную информацию можно найти в документации на сайте технического комитета по стандартизации [1]. Пусть V_s – множество всевозможных двоичных строк длины s . Алгоритм

© Dorokhin S. V., Kachkov S. S., Sidorenko A. A. 2018

© Федеральное государственное автономное образовательное учреждение высшего образования «Московский физико-технический институт (государственный университет)», 2018

зашифрования сводится к последовательному применению следующих операций:

$X[k](a) = k \text{ xor } a$, где K , a прин. V128 – побитовая операция сложения по модулю 2;

$S(a) = S(a_{15}||\dots||a_0) = \text{pi}(a_{15})||\dots||\text{pi}(a_0)$, a_i прин V8 – биективное нелинейное преобразование;

L : V128 \rightarrow V128 – некоторое линейное преобразование.

С учётом этих обозначений функции зашифрования $E[k]$ и расшифрования $D[k]$ могут быть представлены в следующем виде:

$E_{K1,\dots,K10}(a) = X[k_{10}] * \text{LSX}[K9] \dots \text{LSX}[K2] \text{LSX}[K1](a)$;

$D_{K1,\dots,K10}(a) = X[K1]S^{-1}L^{-1}X[K2] \dots S^{-1}L^{-1}X[K9]S^{-1}L^{-1}X[K10](a)$,

где $K1, \dots, K10$ – раундовые ключи. Эти ключи вырабатываются один раз в начале алгоритма и не оказывают существенного влияния на скорость работы шифра. Процедуру генерации этих ключей можно также найти на сайте комитета по стандартизации [1]. Линейное преобразование L может быть осуществлено с помощью 16 циклов работы РСЛОС (регистра сдвига с линейной обратной связью), показанного на рисунке 1.

рисунк 1, регистр сдвига

Такое решение является предпочтительным при аппаратной реализации шифра, однако при создании программной реализации удобнее представить его в матричной форме. Результат работы одного такта РСЛОС можно записать в виде

формулка

Соответственно, результат работы после k тактов:

формулка

Применение L -преобразования в такой форме существенно быстрее, чем при моделировании РСЛОС. Однако профилирование программы при помощи утилиты callgrind показало, что на L преобразование приходится приблизительно 75% времени исполнения программы. Следуя принципу *make common case fast*, необходимо оптимизировать именно L -преобразование, чему, по существу, и посвящена эта статья.

2. Построение таблиц предвычислений

Описать структуру и генерацию LUT.

3. Реализация с использованием векторных инструкций

3.1. Набор инструкций SSE

SSE добавить описание команд, посчитать суммарный CPI и latency, используя интеловский онлайн-справочник

3.2. Учёт особенностей планировщика

2 AGU, чет-нечёт найти какую-нибудь информацию о кол-ве AGU в интеловских процах в таком виде, чтобы на неё можно было сослаться

3.3. Набор инструкций AVX

описать неудачную попытку использовать `ymm` регистры внутри `ApplyXSL`

4. Использование AVX2 в некоторых режимах

Можно заюзать `ymm` там, где нет истинных зависимостей

4.1. Модификация режима ЕСВ

4.2. Модификация расшифрования в режиме СФВ

5. Сравнительный анализ

Зависит от конкретного процессора! Зависит от архитектуры! (skylake и далее - лучше, РАЗОБРАТЬСЯ, ПОЧЕМУ) Посчитать cpb (clocks per byte)

6. Заключение

Текст заключения.

Литература

1. Шевченко Д. В., Шевченко В. П. Выбор и оптимизация структуры построения автономных сейсмических средств обнаружения рубежного типа // Материалы VIII все-российской научно-технической конференции «Современные охранные технологии и средства обеспечения комплексной безопасности объектов». — 2010. — С. 128–133.
2. Diallo M. S., Kulesh M., Holschneider M., Sherbaum F., Adler F. Characterization of polarization attributes of seismic waves using continuous wavelet transforms // Geophysics. — 2006. — V. 71, N. 3. — P. 67–77.
3. Лайонс Р. Цифровая обработка сигналов. — М.: Бином, 2006. — С. 361–369.

References

1. Shevchenko D. V., Shevchenko V. P. Selection and optimization of constructing autonomous seismic detection struction a landmark type // Proceedings of the VIII Russian scientific conference “Modern security technology and means of complex security objectives”. — 2010. — P. 128–133. — (in Russian).
2. Diallo M. S., Kulesh M., Holschneider M., Sherbaum F., Adler F. Characterization of polarization attributes of seismic waves using continuous wavelet transforms // Geophysics. — 2006. — V. 71, N. 3. — P. 67–77.
3. Lyons R. Digital signal processing. — M.: Binom, 2006. — P. 361–369. — (in Russian).

Поступила в редакцию дд.мм.гггг.

Сведения об авторах статей

(на момент подачи статьи)

Реализация шифра ГОСТ Р 34.12 2015 с использованием SIMD инструкций

Дорохин Семён Владимирович (нет, студент 4 курса, Московский физико-технический институт, студент 4 курса) dorohin.sv@phystech.edu

Качков Сергей Сергеевич (нет, студент 4 курса, Московский физико-технический институт, студент 4 курса) kachkov.ss@phystech.edu

Сидоренко Антон Андреевич (нет, студент 4 курса, Московский физико-технический институт, студент 4 курса) sidorenko.aa@phystech.edu

Ссылки на опубликованные статьи (в соответствии с ГОСТ Р 7.0.5-2008)

Дорохин С.В., Качков С.С., Сидоренко А.А. Реализация шифра ГОСТ Р 34.12 2015 с использованием SIMD инструкций // Труды МФТИ. — 2018. — Т. 10, № 4. — С. 1–3.

Dorokhin S.V., Kachkov S.S., Sidorenko A.A. Implementation of GOST R 34.12 2015 cipher using SIMD instructions // Proceedings of MIPT. — 2018. — V. 10, N 4. — P. 1–3.