Санкт-Петербургский государственный университет
Метаматико-Механический Факультет
Кафедра Системного Программирования

Теодам Александр Андреевич

Курсовая работа

Реализацияна программируемой логической интегральной схеме оптимизированного алгоритма атаки на алгоритм шифрования KeeLoq.

Направление 02.04.03
Математическое Обеспечение и
Администрирование Информационных Систем

Научный руководитель,
доктор физ.-мат. наук,
профессор Терехов А. Н.

Заведующий кафедрой,
доктор физ.-мат. наук,
профессор Терехов А. Н.

Санкт-Петербург
2015

Введение

Благодаря развитию беспроводных технологий большинство систем управления в современныхрадиоэлектронных устройствах являются дистанционными – охранные системы для автомобилей, системы ограничения доступа в помещения, идентификационные системы и т.д.

Простые и недорогие системы дистанционного управления используют однонаправленный канал связи, что не может гарантировать надежную защиту и приводит к снижению безопасности системы в целом. В таких устройствах кодовая комбинация не изменяется или их число ограничено. Системы с двунаправленным (обратным) каналом связи имеют высокий уровень защищенности от постороннего вмешательства, но из-за своей сложности и высокой стоимости не нашли широкогокоммерческого применения.

«Взлом» систем с однонаправленным каналом связи и ограниченным числом кодовых комбинаций возможен за короткий промежуток времени, простым перебором всех возможных вариантов. Для таких атак используют сканер кода. Например, в устройствах содержащих восемь конфигурационных перемычек (256 комбинаций) отвечающих за выбор кода защиты, код может быть подобран практически моментально.

Другой способ получения несанкционированного доступа к системе – это использование устройства перехватчика кода. После нажатия кнопки на пульте дистанционного управления кодер передает в эфир кодовую последовательность. Устройство перехвата кода принимает и запоминает данные. Затем, при необходимости, кодовая комбинация повторяется, что приводит к несанкционированному доступу в систему.

Для предотвращения подобных атак в середине 80-х Джидеоном Куном в Nanoteq Pty Ltd был разработан алгоритм шифрования KeeLoq,основанный на функции сдвига регистра, который был продан Microchip Technology Inc в 1995 году. Большинство автопроизводителей используют в своих системах

дистанционного управления KeeLoq – Chrysler, Volvo, Daewoo, GM, Fiat, Honda, Toyota, Jaguar, Volkswagen[1].

Оптимизация для высокопроизводительных видеокарт

В ходе выполнения бакалаврской выпускной квалификационной работы оптимизированный алгоритм атаки была запрограммированна версия для графических процессоров, с помощью технологии Nvidia Cuda. Разработка велась на устаревшей видеокарте Nvidia GeForce 9600 GT(технические характеристики приведены в табл. 1).

Технические характеристики видеокарты Nvidia GeForce 9600 GT	
Число универсальных процессоров	64
Объем видеопамяти	512 Мб
Разрядность шины видеопамяти	256 бит
Частота графического процессора	650 МГц
Частота видеопамяти	1800 МГц
Версия Compute Capability	1.1
Дата выхода	02.2008

Табл. 1

Полученная в результате реализация была запущенна на более производительных и современных видеокартах (технические характеристики приведены в табл. 2), и были достигнуты результаты приведенные в таблице 3.

	Titan Z	Tesla K40	GTX690
Число универсальных процессоров	2880	2496	3072
Объем видеопамяти	6144 Мб	12288 Мб	4096 Мб
Разрядность шины видеопамяти	384 бит	384 бит	512 бит
Частота графического процессора	889 МГц	745 МГц	915 МГц
Частота видеопамяти	7000 МГц	6000 МГц	6008 МГц
Версия Compute Capability	3.5	3.5	3.0
Дата выхода	05.2013	10.2013	05.2012

Табл.2

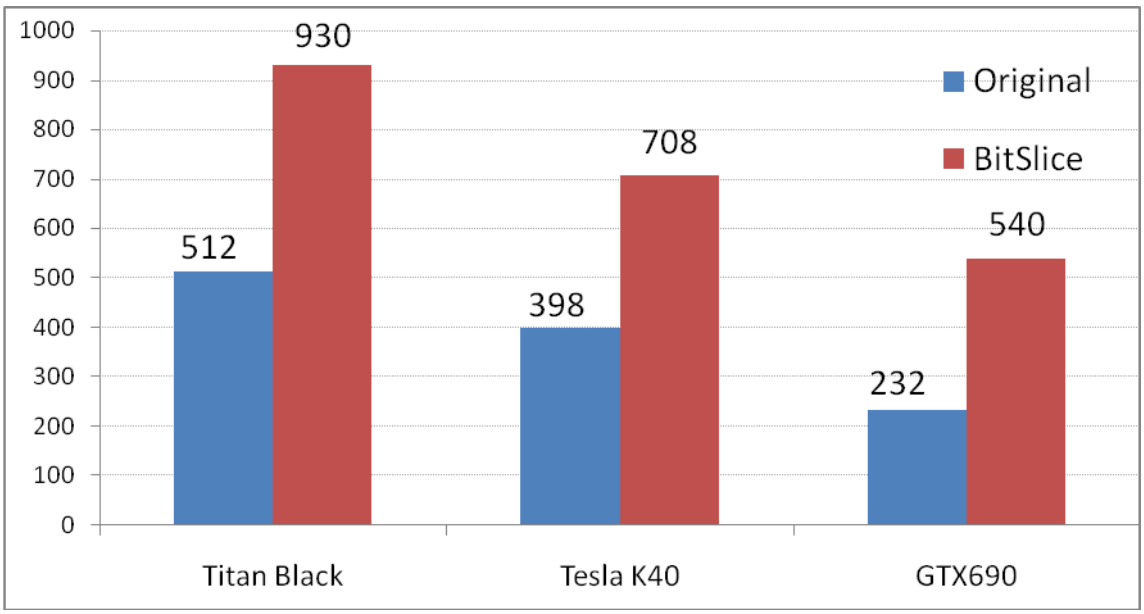

Табл.3

После более тщательной проверки было обнаружено, что более мощные видеокарты, в отличии от слабых, не задействовали все доступные ресурсы. С помощью программы GPU-Z, которая отслеживает состояние видеокарты (рис. 1), было замечено, что графический адаптер используется не на 100%.

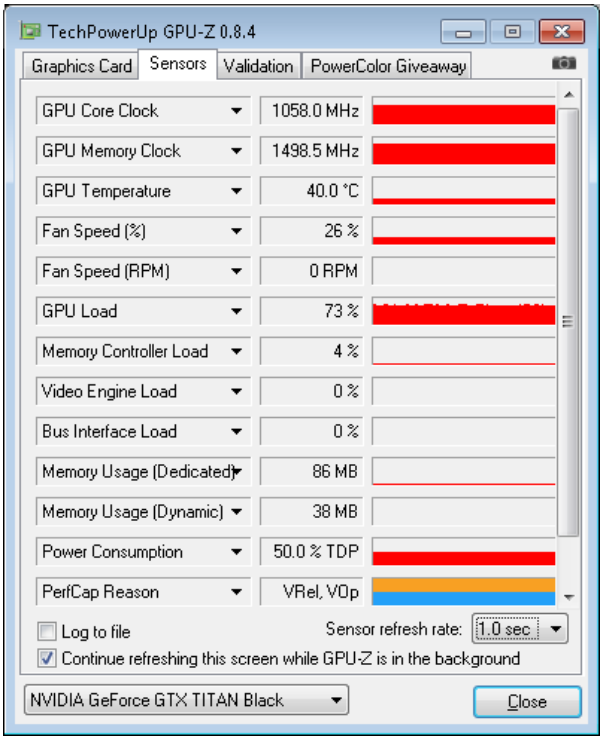

Рис. 1

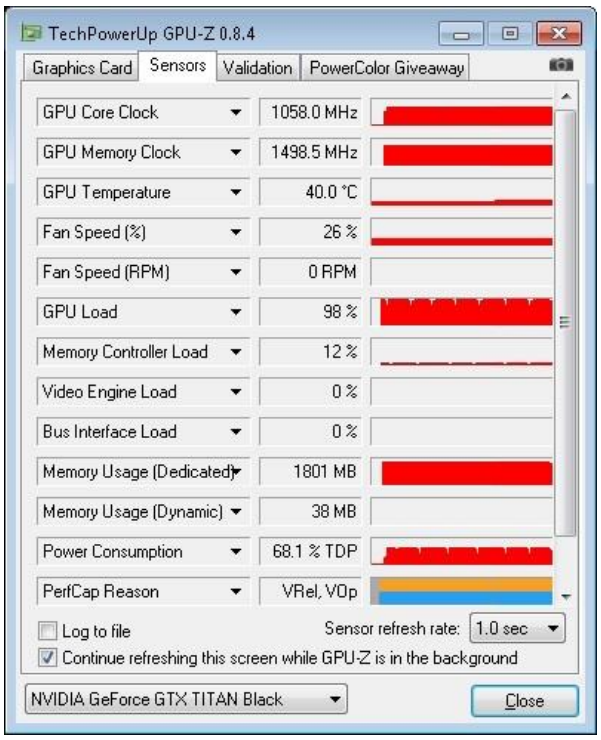

Рис. 2

Возможной причины такого поведения могло быть то что на более мощных видеокартах ядро выполняется гораздо быстрее а, его запуск на занимает некоторое время, из-за частого запуска графический адаптер мог часть времени просто простаивать. Так же было отмечено что видеопамять практически не используется. Для решения этой проблемы программа была переписана, с возможность максимального использования памяти видеокарты, перед запуском ядра, программа определяет сколько «задач» способна рассчитать видеокарта.

Данная модификация позволила полностью загрузить производительные видеокарты (рис. 2) и дала значительный прирост в производительности (табл. 4).

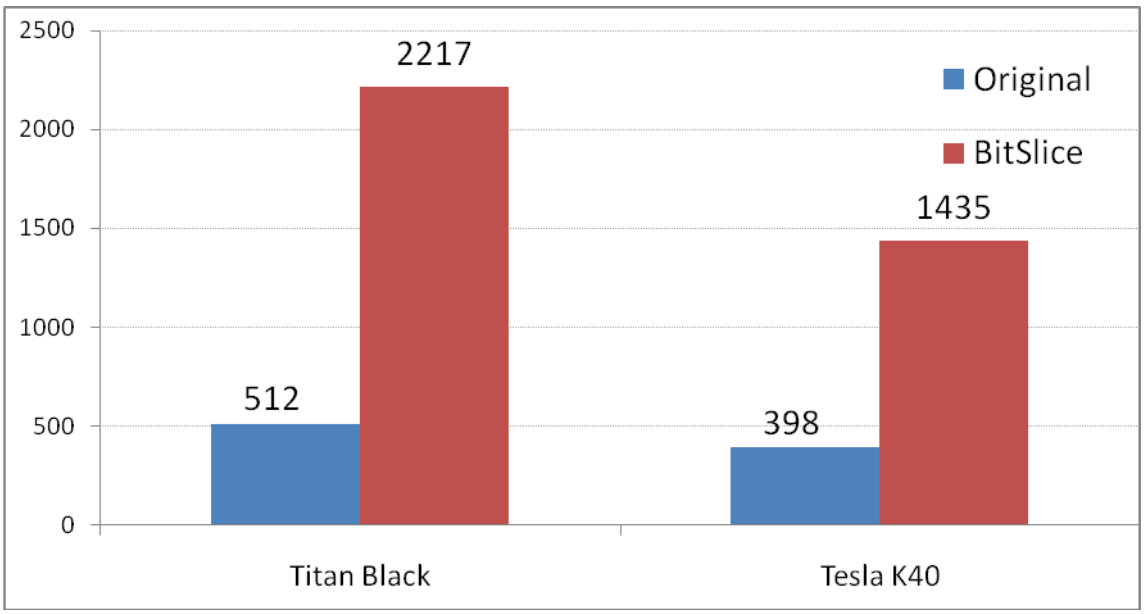

Табл.4

Применение ПЛИС

ПЛИС - это электронные компоненты, предназначенные для дальнейшего создания цифровых устройств. В отличие от обычных интегральных схем, логика работы ПЛИС не определена при изготовлении, она задается в дальнейшем, посредством специальных языков программирования - языков описания аппаратуры, и так же может быть изменена. ПЛИС, наданный момент, являются одними из самых производительных вычислительных устройств.

Применение ПЛИС рассматривались в работах «Wool Cryptanalysis of KeeLoq code-hopping using a Single FPGA» [10] и «Cryptanalysis of KeeLoq with COPACOBANA» [11]. Хотя конечная цель данных работ немного отлична от преследуемой в данной работе, с помощью их результата возможно оценить целесообразность применения этих вычислителей.

В обоих работах предполагается наличие исзвестного серийного номера партии устройство, к которомму добавляется случайное зерно (Random Seed), которое вносит различие в отдельные устройства. Зерно может быть 32, 48, 60 битным, и дополняться оставшимися 32, 16, 4 битами до полного 64 битного ключа, с помощью которого происходит шифрование (рис. 3).

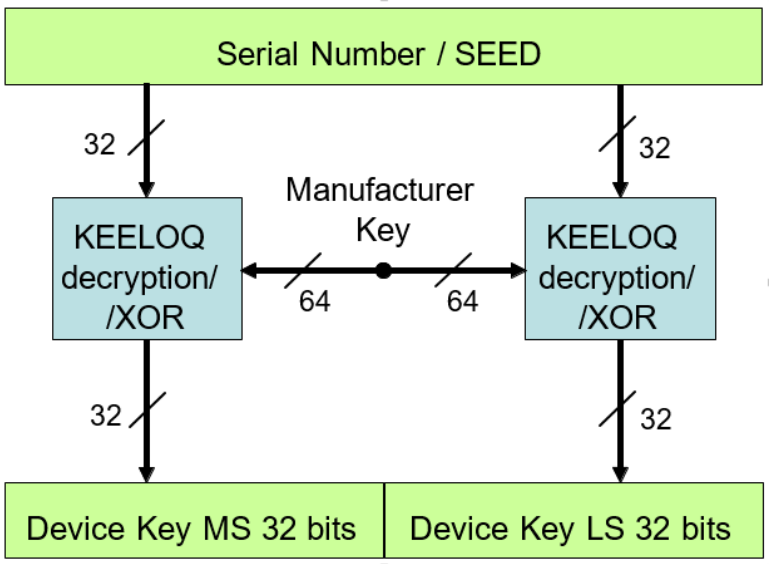

Рис. 3

В [11] рассматривается реализация атаки перебором (BruteForce), на устройстве COPACOBANA (Cost-Optimized Parallel COde Breaker) - вычислитель содержащий 120 ПЛИС Xilinx Spartan3-1000 выполняющих паралельные вычисления. В результате мощность устройства удалось использовать на 83% на частоте 110 МГц, где каждая ПЛИС проверяла 110 миллионо ключей в секунду. Достигнутые результаты приведены в табл. 5

SEED length	1 FPGA	1 COPACOBANA	100 COPACOBANAs
(bits)	(< 80 $)	(< 10000 $)	(< 1000000 $)
32	39 secs	0.33 secs	3.3 msecs
48	29.6 days	5.9 hours	213 secs
60	332 years	1011 days	10.1 days

табл. 5

В работе [10] рассматриваются результаты [11] и проводятся аналогичные эксперименты уже с использованием таких ПЛИС как: Virtex4-100-12 и Virtex6-760-2. Полученные результаты сравниваются с вычислителем COPACOBANA в рассчете ключа с 48 битным зерном (рис. 4).

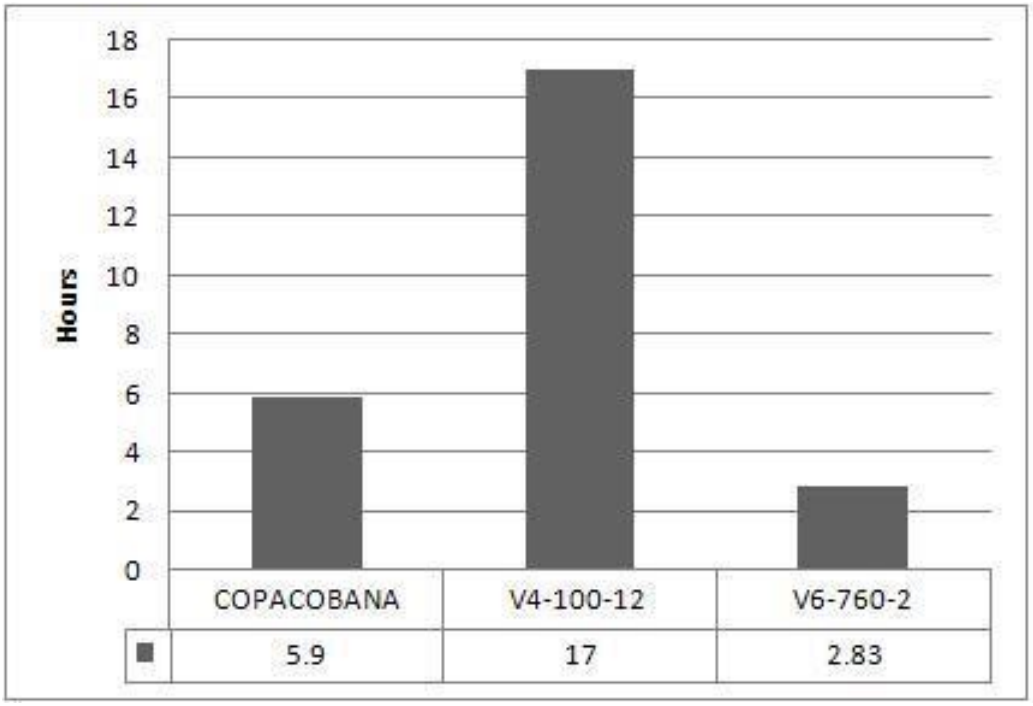

Рис. 4

Результаты этих работ легко сводятся к цели данной курсовой работы если рассмотреть 64 битное зерно, то есть получить полный перебор ключа. Для этого достаточно результаты из табл. 5 умножить на 2^4. Хотя полученные величины в тысячи дней кажутся большими, они являются вполне приемлемыми по сравнению с временем необходимым для полного перебора на CPU и GPU.

Список использованной литературы

1. Обзор технологии KeeLoq. http://www.microchip.ru/lit/keeloq/keeloq_1.htm
2. Bogdanov A. Cryptanalysis of the KeeLoq block cipher// Cryptology ePrint Archive. 2007.
3. IndesteegeS., KellerN., DunkelmanO., BihamE., PreneelB. A Practical Attack on KeeLoq. //Advances in Cryptology – EUROCRYPT2008. 2008.
4. Шнайер Б. Прикладная криптография. 1994.
5. Microchip Technology Inc., HCS410 KeeLoq Code Hopping Encoder Transponder Data Sheet.http://ww1.microchip.com/downloads/en/DeviceDoc/40158e.pdf
6. Взлом KeeLoq http://dedal.com.ua/stati/90-keeloq-vzlom
7. Biryukov, A., Wagner, D. Slide Attacks. // Knudsen, L.R. (ed.) FSE 1999.1999.
8. Основы CUDA. http://steps3d.narod.ru/tutorials/CUDA-tutorial.html
9. CUDA Occupancy Calculator. http://developer.download.nvidia.com/compute/cuda/CUDA_Occupancy_calculator.xls
10. I. Sheetrit and A. // Wool Cryptanalysis of KeeLoq code-hopping using a Single FPGA.
11. M. Novotny and T. Kasper // Cryptanalysis of KeeLoq with COPACOBANA.