

Алгоритмическое и программное обеспечение

УДК 004.932

ИССЛЕДОВАНИЕ БЫСТРОДЕЙСТВИЯ МОДИФИЦИРОВАННОГО КОДЕРА JPEG НА SoC ARM i.MX233 С ПОДДЕРЖКОЙ КОНТРОЛЯ БИТРЕЙТА

Д.В. Сидоров, А.Н. Осокин*

ОАО «Томское пиво», г. Томск

*Томский политехнический университет

E-mail: osokin@vt.tpu.ru

Актуальность работы обусловлена необходимостью создания алгоритмов сжатия изображений с регулируемым битрейтом для использования при ограниченности вычислительных ресурсов (реализация на промышленных контроллерах), ширины каналов передачи данных, патентных ограничений и т. д.

Цель работы: исследование реализованной на ARM i.MX233 нересурсоемкой модификации стандартного алгоритма сжатия JPEG, которая, в отличие от оригинала, обладает возможностью контролировать битрейт сжатых изображений и позволяет использовать любую стандартную реализацию JPEG-декодера для распаковки изображений.

Методы исследования: в работе использованы методы теории информации, теории алгоритмов, математической статистики и объектно-ориентированного программирования.

Результаты: реализован модифицированный кодер стандарта JPEG для системы на кристалле ARM i.MX233, позволяющий контролировать битрейт сжатых изображений. Проведено его исследование на произвольных изображениях, полученных с Web-камеры. Показано, что предложенный кодер с функцией контроля битрейта сжимает изображения быстрее, чем распространенный свободный аналог *cjpeg*.

Ключевые слова:

Кодер JPEG, алгоритм контроля битрейта, ARM система на кристалле i.MX233, ОС Linux.

Введение

Непрерывное совершенствование норм технологического процесса производства сложных полупроводников позволяет выпускать процессоры и системы на кристалле (SoC), которые характеризуются высокой степенью интеграции, производительностью, низким энергопотреблением и способны решать широкий спектр задач в промышленности и потребительском секторе [1]. Одной из таких задач является создание подсистемы видеонаблюдения и видеорегистрации в существующих системах управления и мониторинга производством на основе SoC. Эта задача требует создания такого программного кодера сжатия изображений, который бы относительно легко встраивался в уже существующие системы (например, в виде программного модуля или службы), использовал минимальное количество ресурсов (процессорное время, ОЗУ и т. п.), работал на возможно большем количестве SoC-систем и предъявлял малые требования к ширине канала передачи данных. Решением такой

задачи может служить модифицированный и реализованный авторами нересурсоемкий кодер JPEG с функцией контроля битрейта, для которого декодером может быть любая реализация JPEG в пределах стандарта [2].

Модификация кодера JPEG

Основная идея предложенной модификации кодера JPEG сводится к предположению о том, что существует близкая к линейной зависимость между количеством бит, необходимых для представления всего сжатого изображения, и количеством бит, необходимых для представления заданного количества сжатых ДКП-блоков изображения. Другими словами, во время поиска необходимого коэффициента квантования (качества) Q при JPEG-сжатии, например, методом дихотомии на отрезке $Q_n \in [5, 100]$ можно ограничиться перекодированием (сжатием) небольшого предварительно заданного количества ДКП-блоков, тем самым значительно снизив вычислительную сложность алго-

ритма (количество обработанных пикселей). Более подробная информация о модификации содержится в работе [3].

Модифицированный кодер реализован на языке C++ в виде клиент-серверного приложения (фоновой службы) для ОС Linux, которое считывает изображения с Web-камеры, сжимает их и передает клиенту через последовательный интерфейс RS-232. Выбор такого способа программной реализации позволяет серверному приложению:

- работать практически на любой архитектуре (ARM, x86-32, x86-64, MIPS), которая способна поддерживать ОС Linux;
- передавать данные по наиболее распространенным сетям связи в промышленности (RS-232, RS-422, RS-485);
- практически не зависеть от архитектуры и программного обеспечения клиента, клиент должен только уметь декодировать полученное изображение согласно стандарту JPEG;
- работать параллельно и независимо от основного программного обеспечения системы управления и мониторинга;
- создать автономную, основанную на SoC цифровую видеокамеру.

Взаимодействие между клиентом и серверным приложением осуществляется согласно следующему алгоритму:

Начало.

1. Клиент через последовательный порт запрашивает сжатое изображение с заданным коэффициентом сжатия.
2. Сервер (фоновая служба) считывает исходное изображение с Web-камеры и сжимает его с заданным клиентом коэффициентом сжатия.
3. По окончании сжатия сервер помещает изображение в ОЗУ (если в ОЗУ было старое изображение, то оно перезаписывается) и сообщает клиенту о размере сжатого изображения и количестве блоков, которыми он может его получить.
4. Клиент, исходя из загруженности канала передачи данных и других критериев, запрашивает изображение фиксированными блоками.
5. Следующее изображения клиент получает аналогичным образом, т. е. повторяя шаги 1–4.

Конец.

Предложенная реализация серверного приложения была скомпилирована, запущена и исследована на собранном авторами испытательном стенде на основе SoC ARM i.MX233.

Испытательный стенд на SoC ARM i.MX233

Семейство RISC процессоров ARM и SoC, построенных на их основе, широко и успешно используются в промышленной автоматике, системах с программируемыми логическими контроллерами и человеко-машинных интерфейсах, т. к. обеспечивают одно из лучших соотношений производительность/энергопотребление [4]. Современные производители полупроводников предлагают широкую номенклатуру ARM процессоров и SoC,

отличающихся по функционалу, производительности и энергопотреблению: от малопроизводительных одноядерных контроллеров семейства ARM7TDMI с частотой 16,8 МГц до двухядерных (четырёхъядерных) SoC семейства CORTEX-A15 с частотой до 2,5 ГГц.

В промышленных системах автоматизации и управления с развитым человеко-машинным интерфейсом широко распространены ARM SoC с частотами ядра от 200 до 800 МГц. Связано это в первую очередь с возможностью этих SoC поддерживать ОС высокого уровня, таких как Windows CE/RT, Linux и обеспечивать полноценный графический интерфейс взаимодействия с пользователем, что позволяет значительно упростить и сократить цикл разработки программного обеспечения [5].

Одним из таких SoC является i.MX233 от компании Freescale [6]. Среди аналогичных решений от других производителей, например семейства ARM9 Sitara от Texas Instruments [5], SPEAr3xx от STMicroelectronics [7], LPC31xx и LPC32xx от NXP [8], SAM9X и SAM9M от Atmel [9], i.MX233 обладает богатым набором периферийных устройств и внешних портов, одним из лучших соотношений цена/производительность, достаточной производительностью для решения широкого круга задач (частота ядра до 454 МГц), высокой энергоэффективностью (потребляемая мощность до 0,5 Вт) и хорошей программной поддержкой от производителя в течение 10 лет со дня выпуска образцов (собственная модификация ядра Linux, программные библиотеки для работы с изображениями и видео и т. п.).

Испытательный стенд на SoC i.MX233 был создан авторами на основе прототипа платы Prospero C100 от фирмы Siqube [10], который может выступать в роли аналога реальных систем управления и мониторинга производством. Прототип Prospero C100 обладает следующими основными характеристиками:

- размер 43×62 мм;
- частота ядра ARM9 400 МГц;
- объем ОЗУ DDR I 64 Мб, частота шины 200 МГц;
- диапазон напряжения блока питания $\pm 7-20$ В;
- microSD разъем для твердотельных накопителей;
- порт LVDS для подключения монитора;
- один USB 2.0 хост порт, обеспечивающий питающий ток до 1,5 А.

Созданный стенд также включает в себя следующий набор периферийных устройств: последовательные порты RS-232, Web-камера и Fast Ethernet адаптер, которые были подключены посредством USB-хаба к шине USB 2.0 (рисунок; штрихпунктирной линией обведен минимальный набор аппаратуры, необходимый для функционирования предложенного кодера).

Стенд управляется ОС Debian Linux 6.0 с ядром 2.6.31, которая загружается с microSD флеш карты.

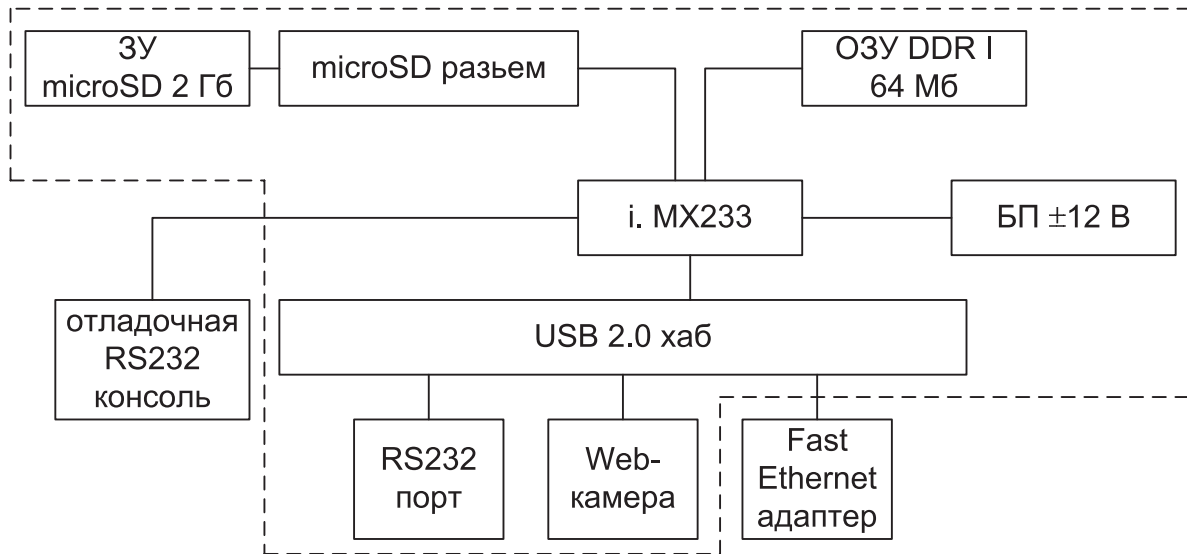


Рисунок. Схема испытательного стенда на SoC i.MX233

Исследование кодера на время сжатия

Целью исследования реализованного на i.MX233 кодера JPEG является получение количественных характеристик времени сжатия произвольных изображений, полученных с Web-камеры.

Результаты исследования кодера на способность соблюдать заданный битрейт приведены в работе [3], там же подробно рассмотрен вопрос качества распакованных изображений.

Исходные данные для исследования:

- произвольные полутоновые изображения, полученные с Web-камеры, размером 640×480 пикс. с глубиной яркости 8 бит на пикс.;
- количество кодируемых ДКП-блоков $n=7,5\%$;
- коэффициент сжатия $K \in [4,30]$, шаг 1;

Максимальное значение коэффициента сжатия K ограничено 30, т. к. при дальнейшем его увеличении качество распакованных изображений становится неприемлемым. Процент сжимаемых ДКП-блоков был взят из работы [3]. Замеры времени сжатия производились как среднее за 500 итераций для каждого коэффициента сжатия. Исходное и сжатые изображения в ходе измерений хранились в ОЗУ. Средство измерения времени – интегрированные часы реального времени с погрешностью порядка 1 мс (время переключения планировщика процессов ОС Linux).

Время сжатия предложенного алгоритма (таблица) сравнивалось с наиболее распространенной свободной реализацией стандарта JPEG–сjreg, которая не обладает возможностью контроля битрейта [11]. Предложенный кодер и сjreg компилировались на стенде одним и тем же компилятором с одинаковыми ключами оптимизации. Размер приложения, занимаемый на твердотельном накопителе, 67 кб, в ОЗУ – 1,2 Мб (для изображения 640×480 пикс. с глубиной 8 бит на пикс.). Параметры сжатия сjreg были подобраны так, чтобы основные этапы сжатия совпадали с аналогичными в

предложенном кодере, т. е. использовалось binDCT и оптимизированные таблицы Хаффмена.

Таблица. Сравнение времени сжатия предложенного кодера с кодером алгоритма сjreg, реализованных на SoC i.MX233

K	Q	Время сжатия (мс)	
		Модифицированного кодера	сjreg реализации
4	86	205	285
5	80	204	284
6	76	205	286
7	66	202	285
8	57	200	282
9	46	201	283
10	42	205	284
11	39	202	281
12	31	200	280
13	28	204	282
14	25	204	282
15	23	202	283
16	21	202	282
17	20	201	282
18	18	201	281
19	17	200	281
20	16	201	281
21	16	202	281
22	14	201	282
23	12	200	281
24	12	201	280
25	11	202	280
26	10	200	280
27	10	201	280
28	10	200	280
29	9	200	280
30	9	200	280
Среднее время		201,7	281,77

Согласно полученным результатам можно сделать следующие выводы: во-первых, кодер может

быть успешно применен на недорогих SoC, поддерживающих ОС высокого уровня; *во-вторых*, он обеспечивает меньшее время сжатия, чем распространенная свободная реализация sjpeg (аналог не обладает функцией контроля битрейта), *в-третьих*, благодаря малому времени сжатия, можно снизить вычислительную нагрузку на процессорный блок SoC и тем самым, например, повысить время автономной работы от аккумуляторных батарей.

Предложенный кодер и его реализация, в отличие от распространенных аналогов, обладает малой вычислительной ресурсоемкостью (меньшим временем сжатия), приемлемой точностью контроля битрейта, не модифицирует схему JPEG-сжатия (распаковка сжатого изображения может быть осуществлена любым декодером JPEG) и, как следствие, может быть с успехом применен при добавлении функций видеонаблюдения и видеорегистрации в существующие системы управления и мониторинга. Благодаря невысоким требованиям предложенной реализации кодера, в систему управ-

ления могут быть дополнительно введены функции автоматической фокусировки/позиционирования камеры на объект, поиска движения в поле зрения камеры и т. д.

Выводы

1. Предложена реализация кодера JPEG, которая в процессе сжатия и контроля битрейта использует малое количество вычислительных ресурсов процессора и может быть успешно использована на недорогих SoC.
2. Проведено исследование кодера на SoC i.MX233 под управлением ОС Linux на произвольных изображениях, полученных с Web-камеры, которое показало, что предложенный кодер с функцией контроля битрейта сжимает изображения быстрее, чем распространенный свободный аналог sjpeg.

Авторы выражают признательность и благодарят главу компании Siquibe Баранова Александра за помощь в сборке, настройке и запуске испытательного стенда.

СПИСОК ЛИТЕРАТУРЫ

1. System-on-chip technology comes of age. 2012. URL: <http://www.eetimes.com/design/power-management-design/4397940/System-on-Chip-technology-comes-of-age> (дата обращения: 12.10.2012).
2. ITU-CCITT Recommendation T.81 Information technology. 1992. URL: <http://www.w3.org/Graphics/JPEG/itu-t81.pdf> (дата обращения: 13.11.2013).
3. Сидоров Д.В., Осокин А.Н. Простой алгоритм контроля битрейта кодера стандарта JPEG // Известия ТПУ. – 2012. – Т. 320. – № 5. – С. 70–73.
4. ARM – The Architecture for the Digital World. 2012. URL: <http://www.arm.com> (дата обращения: 13.10.2012).
5. Обзор платформ цифровых сигнальных процессоров и микропроцессоров на базе ядра ARM компании Texas Instruments. 2012. URL: http://www.ti.com/ww/ru/sitara_whatsnew.html (дата обращения: 10.12.2012).
6. i.MX233 Product Summary Page. 2012. http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=i.MX233 (дата обращения: 27.12.2012).
7. SPEAr Embedded Microprocessors – STMicroelectronics. 2012. URL: <http://www.st.com/internet/mcu/subclass/1156.jsp> (дата обращения: 12.12.2012).
8. ARM9, NXP Semiconductors. 2012. URL: <http://www.nxp.com/products/microcontrollers/arm9/#products> (дата обращения: 24.12.2012).
9. ARM based microcontrollers. 2013. URL: <http://www.atmel.com/products/microcontrollers/arm/default.aspx> (дата обращения: 8.03.2013).
10. ООО «ЭК-Мобайл» Prospero C100. 2012. URL: <http://www.ec-mobile.ru/47prospero/> (дата обращения: 4.03.2012).
11. Independent JPEG Group. 2013. URL: <http://www.iijg.org> (дата обращения: 13.05.2013).

Поступила 11.09.2013 г.

UDC 004.932

RESEARCH FOR COMPRESSING TIME OF MODIFIED JPEG ENCODER WITH BIT RATE CONTROL ON ARM-BASED SoC i.MX233

D.V. Sidorov, A.N. Osokin*

Tomskoe pivo, Tomsk

*Tomsk Polytechnic University

The urgency of the discussed issue is caused by the necessity to develop the algorithms for image compressing with the possibility to use them in the context of limited computational resources, bandwidth of data channels, patent restrictions, etc.

The main aim of the study is to research the implementation on ARM-based SoC i.MX233 of modified JPEG encoder which allows controlling bit rate and using any standard JPEG decoder to decode compressed images.

The methods used in the study: methods of information theory, theory of algorithms, mathematical statistics and object-oriented programming.

The results: the proposed a modification of the standard JPEG encoder for compressing gray-scale images with bit rate control was implemented. The encoder has a simple program implementation, low computational complexity and, as a consequence, low compression time. A study for measuring the compression time of arbitrary gray-scale images from Web-camera on the SoC i.MX233 was conducted on the encoder. The results of the study show that the encoder proposed compresses the images faster than the open source analog cjpeg.

Key words:

JPEG coder, bit rate control, ARM SoC i.MX233, OS Linux.

REFERENCES

1. *System-on-chip technology comes of age*. 2012. Available at: <http://www.eetimes.com/design/power-management-design/4397940/System-on-Chip-technology-comes-of-age> (accessed 12 October 2012).
2. *ITU-CCITT Recommendation T.81 Information technology*. 1992. Available at: <http://www.w3.org/Graphics/JPEG/itu-t81.pdf> (accessed 13 November 2013).
3. Sidorov D.V., Osokin A.N. Prostoy algoritm kontrolya bitreyta kodera standarta JPEG [Simple bit rate algorithm for JPEG encoder]. *Bulletin of the Tomsk Polytechnic University*, 2012, vol. 320, no. 5, pp. 70–73.
4. *ARM – The Architecture for The Digital World*. 2012. Available at: <http://www.arm.com> (accessed 13 October 2012).
5. *Obzor tsifrovyykh signalnykh protsessorov i mikroprotsessorov na baze yadra ARM kompanii Texas Instruments* [Review of ARM-based digital signal processors and microprocessors from Texas Instruments]. 2012. Available at: http://www.ti.com/ww/ru/sitara_whatsnew.html (accessed 10 December 2012).
6. *i.MX233 Product Summary Page*. 2012. http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=i.MX233 (accessed 27 December 2012).
7. *SPEAr Embedded Microprocessors – STMicroelectronics*. 2012. Available at: <http://www.st.com/internet/mcu/subclass/1156.jsp> (accessed 12 December 2012).
8. *ARM9. NXP Semiconductors*. 2012. Available at: <http://www.nxp.com/products/microcontrollers/arm9/#products> (accessed 24 December 2012).
9. *ARM based microcontrollers*. 2013. Available at: <http://www.atmel.com/products/microcontrollers/arm/default.aspx> (accessed 8 March 2013).
10. «EK-Mobayl» *Prospero C100*. 2012. Available at: <http://www.ec-mobile.ru/47prospero/> (accessed 4 March 2012).
11. *Independent JPEG Group*. 2013. Available at: <http://www.ijg.org> (accessed 13 May 2013).