

УДК 004.315

Балкин М.Д.

Студент 2 курса магистратуры, ВШ «Информатика и
вычислительная техника»

Санкт-Петербургский Политехнический университет Петра
Великого
Россия, г. Санкт-Петербург

РАЗРАБОТКА И ИССЛЕДОВАНИЕ АППАРАТНОГО MJPEG ЭНКОДЕРА

Аннотация: В статье рассматривается разработка аппаратного Motion JPEG энкодера на базе ПЛИС для сжатия видеопотока в режиме реального времени. Обосновываются преимущества кодека MJPEG для встраиваемых видеосистем, работающих в условиях ограниченных вычислительных ресурсов, памяти и энергопотребления. На базе реализаций кодека проводится исследование и сравнение Register-transfer level и High-level synthesis подходов к аппаратному проектированию.

Ключевые слова: ПЛИС, видеосистемы, кодеки, JPEG, MJPEG, передача видео в реальном времени, RTL, HLS, высокоуровневый синтез.

Annotation: The article discusses the development of a hardware Motion JPEG encoder based on FPGAs for real-time video stream compression. The advantages of the MJPEG codec for embedded video systems operating in conditions of limited computing resources, memory and power consumption are substantiated. Based on the codec implementations, a study and comparison of approaches to hardware design at the level of register transfer and high-level synthesis was conducted.

Key words: FPGA, VIDEO SYSTEMS, CODECS, JPEG, MJPEG, REAL-TIME VIDEO TRANSMISSION, RTL, HLS, HIGH-LEVEL SYNTHESIS.

В современном мире встраиваемые системы, помимо всего прочего, нашли широкое применение в задачах обработки и передачи видео, так, они используются в машинном зрении, промышленной автоматизации, робототехнике, транспортных системах, медицинской аппаратуре и других областях. Во всех этих случаях от системы требуется получить поток видеоданных, провести его первичную обработку и передать по каналу связи или записать в память. Видеоданные занимают большой объем памяти даже при низком разрешении и небольшом количестве кадров в секунду, а при использовании форматов высокой четкости объем занимаемой памяти кратно увеличивается. Так, для уменьшения объема видеоданных используются кодеки, которые кодируют и декодируют видеопоток при помощи алгоритмов сжатия.

В контексте встраиваемых видеосистем выбор подходящего кодека играет особую роль, так как системе предстоит работать в условиях жестких ограничений по вычислительной мощности, доступной памяти и энергопотреблению. Помимо этого, нужно обращать внимание на вычислительную сложность кодека и создаваемые им задержки обработки, так как для систем реального времени нарушение временных ограничений может привести к потерям кадров. Таким образом, при проектировании встраиваемой видеосистемы необходимо выбрать кодек, сочетающий удовлетворительную степень сжатия с допустимой вычислительной сложностью и предсказуемыми временными характеристиками [3, 4].

В работе проводится разработка кодека Motion JPEG, основанного на внутрикадровом сжатии. Его принцип работы заключается в том, что видеопоток представляется как набор независимых кадров, каждый из которых кодируется алгоритмом JPEG. В отличие от современных межкадровых кодеков H.264 и H.265, использующих временное предсказание и зависимости между соседними кадрами, MJPEG не анализирует межкадровые изменения и не использует опорные кадры. Такой подход уменьшает вычислительную сложность кодека, требует намного меньше

ресурсов и обеспечивает предсказуемые задержки обработки кадров, хотя и обладает более низким коэффициентом сжатия [4].

Алгоритм JPEG относится к категории алгоритмов сжатия с потерями и использует особенности восприятия изображения человеческим глазом, что позволяет сократить объем данных за счёт контролируемых потерь в тех компонентах сигнала, которые мало влияют на визуальное качество. Так, работа алгоритма начинается с преобразования цветового пространства изображения из RGB, в котором данные поступают с видео захвата, в формат YUV, в котором яркостный и два хроматических канала разделены. Это необходимо для проведения этапа субдискретизации, где для каждой отдельной группы пикселей 4x2 цветное разрешение снижается, а яркостное сохраняется. Далее, изображение разбивается на блоки размером 8x8 пикселей, для каждого из которых выполняется дискретное косинусное преобразование, переводя данные из пространственной области в частотную, что позволяет выделить коэффициенты, соответствующие низким и высоким пространственным частотам. На этапе квантования полученные значения делятся на элементы матрицы квантования и округляются. В результате высокочастотные составляющие, мало влияющие на визуальное восприятие изображения, уменьшаются или полностью обнуляются, за счет чего и достигается основное сжатие. На заключительном этапе выполняется энтропийное кодирование, включающее алгоритмы RLE и Хаффмана, благодаря которым объем информации уменьшается без потерь [2, 4].

В качестве аппаратной платформы для построения встраиваемых видеосистем часто используются программируемые логические интегральные схемы (ПЛИС), так как на их основе можно эффективно реализовывать тракты потоковой обработки данных в режиме реального времени. В задачах обработки и сжатия видео основные вычисления представляют собой набор повторяющихся операций над пикселями и блоками изображения, что делает такие алгоритмы удобными для распараллеливания и конвейеризации. Так, разные стадии работы энкодера могут одновременно выполняться над

разными частями входного потока, что обеспечивает высокую пропускную способность. Так же, при описании устройства на ПЛИС можно заранее оценить задержки обработки данных на том или ином этапе, что важно при работе с системами реального времени. В работе реализация Motion JPEG энкодера осуществляется на базе ПЛИС Xilinx семейства Artix-7 [3].

В настоящий момент на рынке IP ядер (логический модуль, готовый для внедрения) уже существуют решения как отдельного JPEG ядра, так и полного аппаратного MJPEG тракта. Такие решения представлены в таблице 1, среди них единственным бесплатным является решение от компании Gowin, которое можно использовать только на ПЛИС их производства. Остальные IP ядра коммерческие и предоставляются сторонними поставщиками. Помимо того, что все эти решения платные или применимы только к конкретным ПЛИС, в готовое IP ядро невозможно внести изменения, как и посмотреть исходный код, что может быть критичным при отладке конечного устройства.

Таблица 1. Конкуренционные решения

Вендор	IP ядро	Тип IP ядра	Разрешение
Gowin[7]	MJPEG Encoder IP	MJPEG-ядро	от 64×64 до 1920×1080
CAST[8]	JPEG-EX-S	JPEG-ядро	до 1920×1080 (Full HD)
CAST[9]	Motion JPEG Over IP	Полный тракт MJPEG	до 1920×1080 (Full HD)
Alma Technologies [10]	JPEG-E	JPEG-ядро	от 1920×1080 до UHD
Parretto [11]	JPEG Encoder IP	JPEG-ядро	от 1920×1080 до 4K

Таким образом, в рамках работы одной из целей является разработка свободно распространяемого IP-ядра Motion JPEG энкодера с открытым исходным кодом в разрешении Full HD и частотой 25 кадров в секунду. Другой же целью является сравнение эффективности двух подходов к аппаратному проектированию на базе разработки Motion JPEG энкодера: классического Register-Transfer Level (RTL) и нового метода High-Level Synthesis (HLS).

Аппаратное проектирование цифровых устройств представляет собой процесс разработки цифровой логики, которая может быть реализована на базе ПЛИС или в виде ASIC (Интегральная Схема Специального Назначения). В

отличие от программной реализации, где алгоритм выполняется процессором как последовательность команд, при аппаратном проектировании создается структура самого вычислительного устройства: регистры, комбинационная логика, блоки памяти, управляющие автоматы и интерфейсы обмена данными. Обычно для такого рода проектирования используется Register-Transfer Level (RTL) подход, который предполагает описание структуры цифрового устройства на уровне регистровых передач при помощи языков описания аппаратных средств VHDL, Verilog и SystemVerilog. Так, разработчик описывает поведение устройства на уровне тактовых процессов, то есть определяет, как данные хранятся в регистрах на каждом такте, как и при каких условиях данные передаются, как организован управляющий конечный автомат, где используется комбинационная логика, а где синхронная [5]. Такой подход обеспечивает высокий уровень контроля над архитектурой конечного устройства и позволяет проводить тонкую оптимизацию по быстродействию, задержкам и аппаратным ресурсам вплоть до битов, однако этот процесс очень трудоёмкий и требует от разработчика квалификации.

Альтернативой является подход High-Level Synthesis (HLS), основанный на автоматической генерации аппаратной реализации из алгоритмического описания на более высоком уровне абстракции. Так, поведение устройства описывается на высокоуровневом языке SystemC, после чего инструмент HLS синтезирует RTL описание, которое реализуется на ПЛИС. При этом разработчик в основном оперирует последовательностью вычислений, структурами данных и зависимостями между этапами выполнения алгоритма, а формирование регистровой структуры и управляющих автоматов происходит автоматически. Для управления архитектурой и оптимизации используются директивы синтеза, при помощи которых можно проводить конвейеризацию, развёртку циклов, организацию памяти и создавать интерфейсы [1]. Такой подход менее трудоёмкий, требует меньшей квалификации от разработчика и позволяет быстрее получить рабочий прототип устройства, однако финальная реализация сильно зависит от

примененных директив синтеза и возможности распараллеливания исходного алгоритма.

Сравнение реализаций MJPEG-энкодера, полученных двумя разными методами аппаратного проектирования, будет проводиться по нескольким группам метрик. К первой группе относятся показатели качества и эффективности кодирования: коэффициент сжатия кадра и среднеквадратичная ошибка, отражающая разницу восстановленного и исходного изображений. Вторая группа связана с быстродействием и включает оценку задержки между поступлением видеопотока на вход энкодера и формированием полного сжатого кадра на выходе [6]. Третья группа характеризует аппаратные затраты ресурсов ПЛИС, в частности количество таблиц соответствия (LUT), триггеров и блоков памяти BRAM. Четвёртая группа содержит метрики трудоёмкости и времени затраченного на разработку.

В конечном итоге лучшая из полученных реализаций будет использована в полном тракте видеопередачи, изображенном на рисунке 1. Он включает в себя модуль видеозахвата, передающий RGB видеопоток, аппаратный Motion JPEG энкодер, подсистему передачи данных через Ethernet и принимающее устройство.

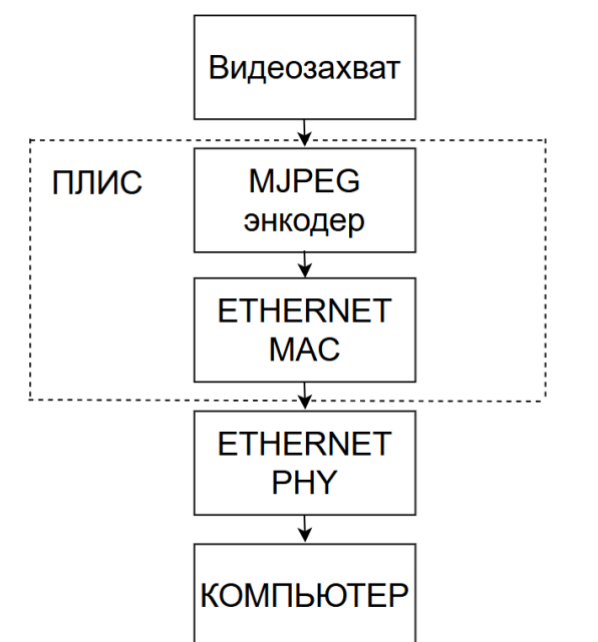


Рис. 1. Схема видеотракта

ЛИТЕРАТУРА:

1. Getting Started with Vitis HLS [Электронный ресурс]. – URL: <https://docs.amd.com/r/2021.2-English/ug1399-vitis-hls> (дата обращения 01.03.2026)
2. Digital compression and coding of continuous-tone still images [Электронный ресурс]. – URL: <https://www.itu.int/rec/T-REC-T.81-199209-I/en> (дата обращения 01.03.2026)
3. Embedded Computer Vision – Branislav Kisacanin, Shuvra S. Bhattacharyya [Текст], 2009, 281 с. Springer. – С.101-111
4. Digital Video Processing for Engineers – Suhel Dhanani, Michael Parker [Текст], 2012, 224 с. Newnes. – С.5-9, 69-72, 97-108
5. Vivado Design Suite User Guide: Synthesis [Электронный ресурс]. – URL: <https://docs.amd.com/r/en-US/ug901-vivado-synthesis/> (дата обращения 01.03.2026)
6. Subjective video quality assessment methods for multimedia applications [Электронный ресурс]. – URL: <https://www.itu.int/epublications/publication/itu-t-p-910-2023-10-subjective-video-quality-assessment-methods-for-multimedia-applications> (дата обращения 01.03.2026)
7. Gowin MJPEG Encoder IP User Guide [Электронный ресурс]. – URL: https://www.gowinsemi.com/upload/database_doc/2289/document/627030718d732.pdf (дата обращения 01.03.2026)
8. JPEG-E-S Baseline JPEG Encoder [Электронный ресурс]. – URL: <https://www.cast-inc.com/compression/jpeg-image-compression/jpeg-e-s> (дата обращения 01.03.2026)
9. Motion JPEG Over IP – HD Video Encoder Subsystem [Электронный ресурс]. – URL: https://www.cast-inc.com/sites/default/files/pdfs/2020-02/cast_mjpegoip-hde.pdf (дата обращения 01.03.2026)

10. Alma Technologies JPEG-E [Электронный ресурс]. – URL: <https://www.alma-technologies.com/ip-core.JPEG-E> (дата обращения 01.03.2026)
11. Parretto JPEG Encoder IP [Электронный ресурс]. – URL: <https://www.parretto.com/jpeg> (дата обращения 01.03.2026)