

СФ-блок кодирования Хаффмана для сжатия изображений по стандарту JPEG

И.А. Беляев

ГУП НПЦ «ЭЛВИС», ioganness@yandex.ru

Аннотация — В данной работе рассматривается аппаратная реализация СФ-блока кодирования Хаффмана, предназначенного для сжатия изображений по стандарту JPEG в составе систем на кристалле (СнК). Приводится процедура кодирования Хаффмана, используемая в стандарте сжатия JPEG. Приведены структура и логика работы СФ-блока кодирования Хаффмана, а также рассмотрены проблемы встраивания данного СФ-блока в СнК.

Ключевые слова — сжатие изображений, JPEG, кодирование Хаффмана, СФ-блок, система на кристалле, СнК.

I. ВВЕДЕНИЕ

На сегодняшний день наиболее распространенным стандартом сжатия цифровых фотографических изображений является стандарт JPEG [1]. Несмотря на то, что формат JPEG требует значительных вычислений, его популярность и выдающиеся возможности по сжатию изображений обычно перевешивают недостаток, связанный с большими затратами времени на обработку данных.

Программная реализация алгоритма сжатия JPEG существует на многих цифровых процессорах. Алгоритм JPEG состоит из нескольких этапов, наиболее затратным по времени из которых является кодирование Хаффмана [3]. Поскольку коды Хаффмана имеют переменную длину, упаковка потока кодов в фиксированные по длине слова памяти связана с постоянными операциями сдвига и проверки переполнения текущего слова памяти. В свою очередь, каждая проверка переполнения текущего слова памяти является операцией условного ветвления, и в случае выполнения условия приводит к перезагрузке программного конвейера и потере нескольких тактов. Кроме того, для процессоров, система команд которых не имеет операций с байтами, задача формирования сжатых данных усложняется определённым в стандарте JPEG байт-стаффингом.

В данной работе рассматривается аппаратная реализация СФ-блока кодирования Хаффмана, который может использоваться в составе СнК в качестве специализированного вычислительного блока при сжатии изображений по стандарту JPEG. В таком случае процедура кодирования Хаффмана

выполняется, во-первых, гораздо быстрее, а во-вторых, параллельно с остальными этапами сжатия изображения, что существенно ускоряет кодирование JPEG в целом. Платой за быстродействие становятся аппаратные затраты, однако для данной реализации СФ-блока они невелики.

II. КОДИРОВАНИЕ ХАУФМАНА В JPEG

Алгоритм Хаффмана (Huffman) — алгоритм префиксного кодирования алфавита с минимальной избыточностью был разработан в 1952 году доктором Массачусетского технологического института Дэвидом Хаффманом.

Идея алгоритма заключается в том, что наиболее часто употребляемым (наиболее вероятным) символам кодируемого алфавита присваиваются наиболее короткие коды.

В стандарте JPEG кодирование Хаффмана имеет дело с двумерным массивом элементов, называемых квантованными коэффициентами. Сам массив обычно называется блоком квантованных коэффициентов или единицей данных [4]. Блок имеет размер 8×8 элементов. Данный массив формируется на этапах сжатия JPEG, предваряющих кодирование Хаффмана, и, по сути, является преобразованным массивом 8×8 пикселей определённой цветовой компоненты исходного изображения. Левый верхний элемент называется коэффициентом DC, остальные элементы — коэффициентами AC. Данный массив обладает тем свойством, что чем более коэффициенты AC удалены от коэффициента DC, тем с большей вероятностью их значение равно нулю. Под удалённостью в данном случае следует понимать сумму индексов коэффициента AC: чем она больше, тем дальше данный коэффициент от DC.

В большинстве случаев, при достаточно высокой степени сжатия изображения, большинство коэффициентов AC оказываются нулевыми. Для того чтобы сгруппировать рядом как можно больше таких коэффициентов, используется процедура зигзаг-сканирования. В результате её выполнения массив 8×8 элементов преобразуется в строку из 64 элементов, как показано на рис. 1. Полученная строка обладает тем важным для кодирования Хаффмана

свойством, что при прохождении по ней слева направо вероятность встретить ненулевой элемент тем ниже, чем больше нулевых коэффициентов уже было встречено.

Кодирование Хаффмана коэффициентов DC и AC полученной строки происходит отдельно и основывается на специальных таблицах кодов Хаффмана.

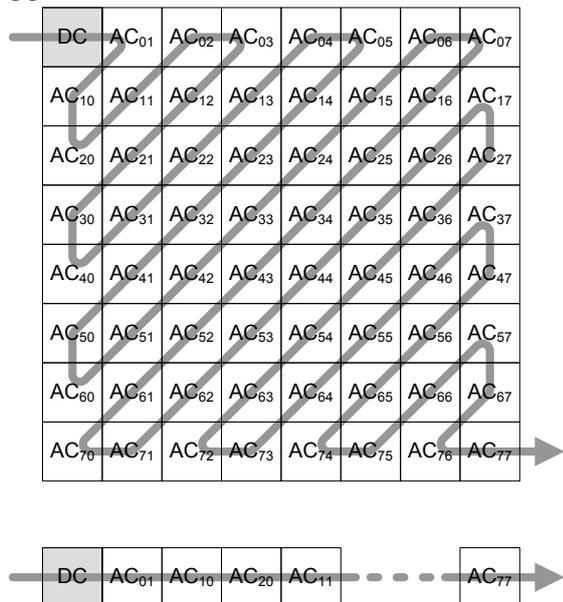


Рис. 1. Зигзаг-сканирование массива квантованных коэффициентов

А. Кодирование коэффициентов DC

Коэффициент DC несет основную часть информации о кодируемом блоке. Так как коэффициент DC имеет большое абсолютное значение, то кодируется не он сам, а разность (DIFF) между ним и так называемым предсказанием коэффициента DC, которым является значение коэффициента DC предыдущего блока.

Номер старшего ненулевого двоичного разряда полученной разности DIFF задает категорию (SSSS) – диапазон, в котором лежит DIFF. Согласно этой категории из определённой таблицы кодов извлекается код Хаффмана.

Так как код Хаффмана указывает только на порядок разности DIFF, то вводятся дополнительные биты, уточняющие категорию до точного значения DIFF. В качестве дополнительных битов берутся SSSS младших битов DIFF. Таким образом, код для разности DIFF состоит из двух частей, как показано на рис. 2.

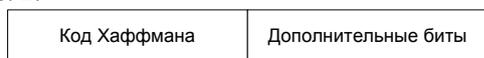


Рис. 2. Структура кода квантованного коэффициента

Б. Кодирование коэффициентов AC

После кодирования коэффициента DC в блоке останется 63 коэффициента AC. Поскольку с большой долей вероятности более половины из них будут равны нулю, кодируются только ненулевые коэффициенты. При проходе по коэффициентам AC создается специальный кодированный байт вида 0xRRRRSSSS.

RRRR (4 бита) – серия нулевых значений, предшествующих первому ненулевому коэффициенту. Если количество нулей в серии равно 16, то все они кодируются специальным кодом, называемым ZRL. Если до конца блока все оставшиеся коэффициенты – нули, то все они кодируются специальным кодом конца блока, называемым EOB.

SSSS (4 бита) – категория ненулевого коэффициента AC.

Таблица кодов Хаффмана для коэффициентов AC организована таким образом, что наиболее короткие коды соответствуют коэффициентам:

- наиболее малым по величине;
- следующим после наиболее короткой серии нулевых коэффициентов.

Структура кода для коэффициентов AC аналогична таковой для коэффициента DC и состоит непосредственно из кода Хаффмана и следующих за ним дополнительных битов.

III. СТРУКТУРА И ЛОГИКА РАБОТЫ СФ-БЛОКА КОДИРОВАНИЯ ХАФФМАНА

Разработанная аппаратная реализация СФ-блока кодирования Хаффмана предназначена для встраивания в СнК «МУЛЬТИКОР» и имеет необходимый для этого набор интерфейсов:

- системный интерфейс;
- АХИ-подобный интерфейс управления;
- АХИ-подобный интерфейс обращений к памяти.

СФ-блок кодирования Хаффмана не имеет ОЗУ для хранения блока квантованных коэффициентов и ПЗУ для хранения таблиц кодов Хаффмана. Все входные данные СФ-блока хранятся в памяти СнК. Это существенно уменьшает размеры СФ-блока, однако приводит к тому, что СФ-блок становится дополнительным абонентом памяти и усложняет логику арбитража обращений к ней.

Структура СФ-блока кодирования Хаффмана приведена на рис. 3.

Для управления кодированием в СФ-блоке предусмотрен набор программно-доступных регистров, приведённый в таблице 1.

Алгоритм работы СФ-блока состоит из следующих этапов:

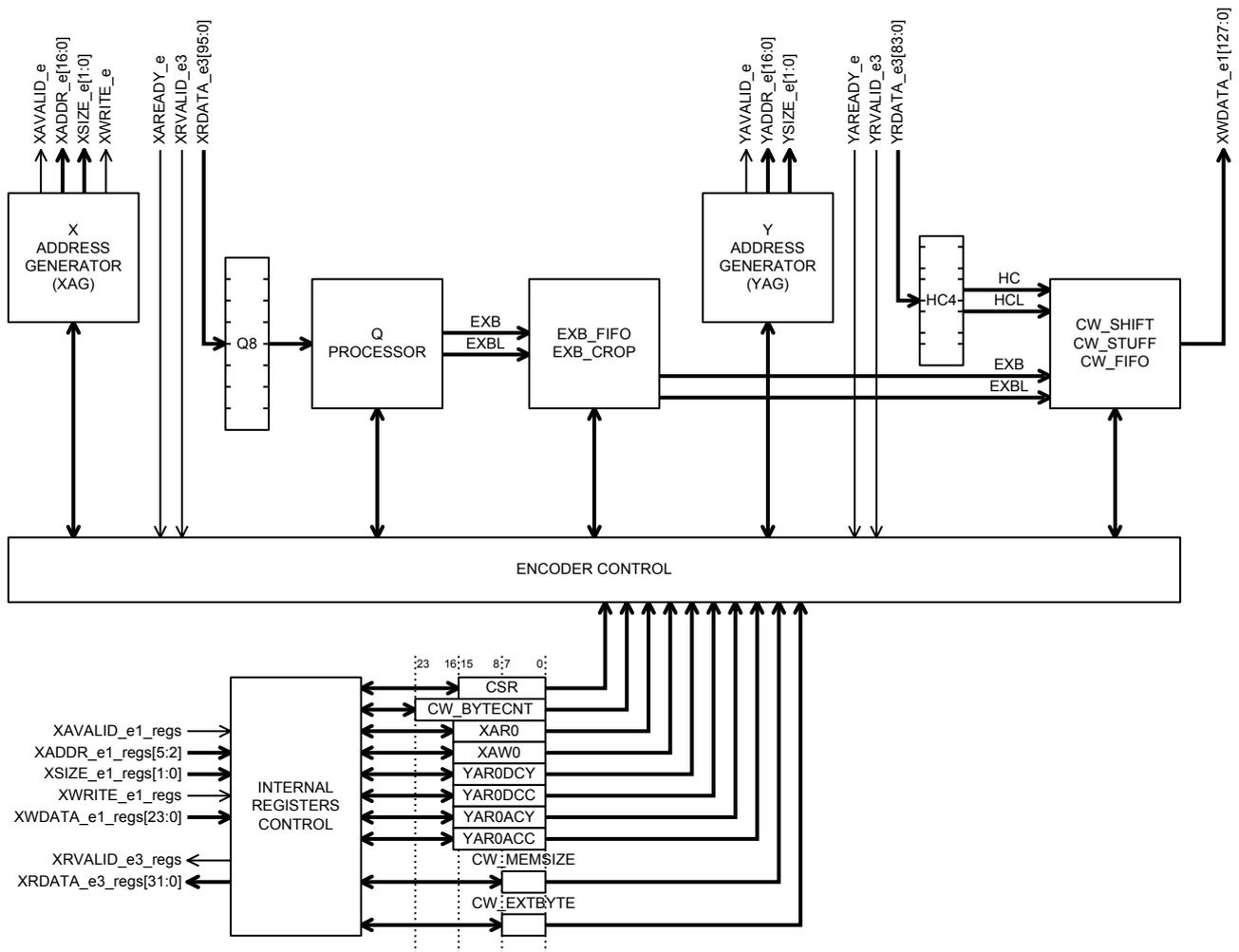


Рис. 3. Структура СФ-блока кодирования Хаффмана

А. Чтение 128-разрядного слова квантованных коэффициентов из памяти

За одно обращение к памяти во внутренней регистр Q8 СФ-блока считывается одно 128-разрядное слово, содержащее 8 квантованных коэффициентов, причем несущественные биты (4 старших бита каждой 16-разрядной части 128-разрядного слова) игнорируются сразу. Адрес и управляющие сигналы формируются блоком XAG.

Б. Анализ и обработка 128-разрядного слова квантованных коэффициентов

На этом этапе осуществляется определение типа коэффициента (DC или AC), определение его категории (SSSS) и длины серии нулей перед ним (RRRR). В блоке EXB_CROP формируются дополнительные биты (EXB), ожидающие в очереди EXB_FIFO прихода из памяти соответствующих им кодов Хаффмана.

В. Чтение из памяти кодов Хаффмана

Адрес и управляющие сигналы для чтения из памяти кодов Хаффмана формируются блоком YAG. Таблица кодов Хаффмана в памяти организована таким образом, что младшие 16 бит 32-разрядного слова памяти содержат код Хаффмана, а старшие – его длину.

Г. Формирование 32-разрядного кодового слова

Формирование 32-разрядного кодового слова осуществляется в блоке CW_SHIFT. Он содержит систему сдвиговых регистров, без задержек обрабатывающих поток кодов Хаффмана (HC) и дополнительных битов (EXB)

Д. Байт-стаффинг 32-разрядного кодового слова

Дальнейшая обработка кодового слова заключается в так называемом байт-стаффинге. Байт со значением 0xFF зарезервирован в стандарте JPEG как идентификатор начала маркера и не может просто так

находиться в сжатых данных. В соответствии со стандартом, в сжатых данных сразу после байта 0xFF необходимо вставить байт со значением 0x00. При декодировании этот байт будет проигнорирован и позволит избежать ошибок ложного появления маркеров. Байт-стаффинг осуществляется в блоке CW_STUFF параллельно для 32-разрядного слова.

Е. Запись 128-разрядного кодового слова в память

Полученные в процессе кодирования сжатые байты пакуются в 128-разрядные слова. Адрес и управляющие сигналы для записи в память кодового слова формируются блоком XAG.

Таблица 1

Программно-доступные регистры СФ-блока кодирования Хаффмана

Название	Размер, бит	Назначение
CSR	16	Регистр управления и состояния
CW_BYTECNT	24	Счетчик байтов данных (кодовых слов), записанных в память
XAR0	17	Адрес начала входного массива данных в памяти (чтение по тракту X)
XAW0	17	Адрес начала области памяти для записи выходных данных (запись по тракту X)
YAR0DCY YAR0DCC YAR0ACY YAR0ACC	17	Адреса начала таблицы кодов Хаффмана в памяти (чтение по тракту Y)
CW_MEMSIZE	8	Размер области памяти для круговой записи выходных данных
CW_EXTBYTE	8	Внешний байт для вставки в выходную последовательность

В результате выполнения вышеописанных этапов в памяти, начиная с адреса XAW0, формируются сжатые данные JPEG.

IV. АППАРАТНЫЕ ЗАТРАТЫ И ПРОИЗВОДИТЕЛЬНОСТЬ

Временные затраты существующей программной реализации сжатия JPEG на базе СнК «МУЛЬТИКОР» приведены в таблице 2.

Таблица 2

Вычислительные затраты отдельных процедур алгоритма сжатия JPEG

Преобразование цветовых компонент	4,0 инстр./пиксел
Дискретное косинусное преобразование	4,1 инстр./пиксел
Квантование	1,1 инстр./пиксел
Зигзаг-сканирование	1,7 инстр./пиксел
Кодирование Хаффмана	4-24, в среднем ~9 инстр./пиксел
Всего	~20 инстр./пиксел

При частоте процессора 300 МГц скорость сжатия JPEG составит не более 5 мегапикселей в секунду.

СФ-блок кодирования Хаффмана можно рассматривать как «дешёвый» в плане аппаратных затрат ускоритель сжатия JPEG, призванный избавить процессор от программного кодирования. Скорость обработки СФ-блоком массивов квантованных коэффициентов составляет ~2 такта на пиксель. Учитывая то, что СФ-блок работает параллельно с основным вычислительным ядром СнК, скорость сжатия JPEG можно оценить как ~11 инстр./пиксел.

За скорость приходится платить площадью, занимаемой СФ-блоком на кристалле. При логическом синтезе СФ-блока на библиотеке элементов с проектными нормами изготовления 0,18 мкм были получены следующие характеристики:

1. Предельная рабочая частота – 450 МГц;
2. Суммарная площадь стандартных логических элементов, входящих в состав СФ-блока, без учета межсоединений – около 190000 мкм², или 0,19 мм²

V. ЗАКЛЮЧЕНИЕ

Занимая в масштабе СнК небольшую площадь, СФ-блок кодирования Хаффмана позволяет почти вдвое ускорить сжатие изображений по стандарту JPEG по сравнению с чисто программной реализацией. Использование этого СФ-блока может быть полезно в СнК, ориентированных на обработку изображений или сжатие видео в формате MJPEG.

ЛИТЕРАТУРА

[1] ITU Recommendation T.81 // ISO/IEC 10918-1 : 1993(E).
 [2] Hamilton E. JPEG File Interchange Format. Version 1.02. – 1991.
 [3] Huffman David A. A Method for the Construction of Minimum-Redudancy Codes. – 1952.
 [4] Миано Дж. Форматы и алгоритмы сжатия изображений в действии. – М.: Издательство Триумф, 2003. – С. 145-166.