

С. И. ЕГОРОВ, В. О. АВДЕЕВ, Э. И. ВАТУТИН, В. С. ПАНИЩЕВ

ДЕКОДЕР LDPC-КОДОВ ДЛЯ ЦИФРОВОГО ТЕЛЕВИДЕНИЯ

Предложена структурно-функциональная организация декодера LDPC-кодов для приемников цифрового телевидения второго поколения, обеспечивающая малые реализационные потери исправляющей способности и уменьшенную аппаратную сложность реализации.

Ключевые слова: DVB-S2, LDPC-код, итеративное декодирование, декодер.

Качество и стоимость телевизионных приемников систем цифрового телевидения второго поколения (DVB-S2 и DVB-T2 [1]) во многом зависят от важного компонента: декодера помехоустойчивых LDPC-кодов. Это делает актуальной задачу разработки декодеров длинных LDPC-кодов с невысокой сложностью аппаратной реализации и высокой исправляющей способностью. Вариант такого декодера предлагается в настоящей работе.

LDPC-код — это линейный блочный код длины n с k информационными и m проверочными символами, задаваемый проверочной матрицей $\mathbf{H}=[h_{ij}]_{m \times n}$, определяющей вид двудольного графа Таннера [2].

Граф Таннера (рис. 1) имеет два множества вершин. Одно состоит из m проверочных вершин $\{c_1, c_2, \dots, c_m\}$, соответствующих m строкам матрицы \mathbf{H} , второе — из n кодовых вершин $\{b_1, b_2, \dots, b_n\}$, соответствующих n столбцам матрицы \mathbf{H} . Кодовая вершина b_j соединяется ребром с проверочной вершиной c_i в том случае, если в ячейке h_{ij} матрицы \mathbf{H} находится единица.

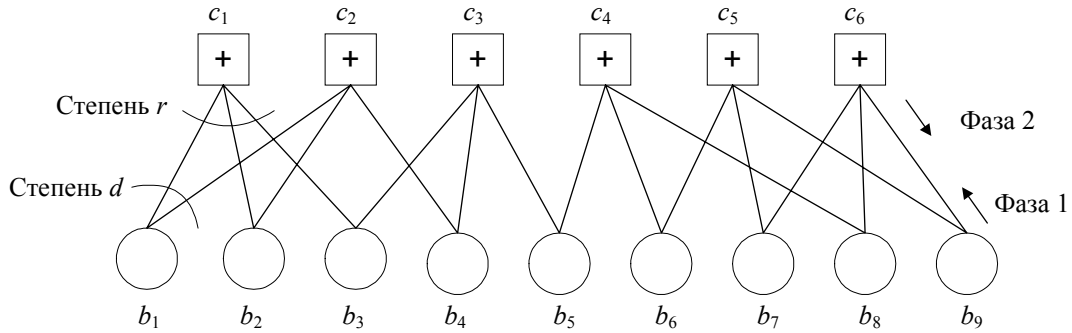


Рис. 1

В стандарте DVD-S2 (DVB-T2) [1] используются два множества структурированных LDPC-кодов: „длинные“ коды (normal) с длиной слова 64 800 бит и „короткие“ коды (short) с длиной слова 16 200 бит. Эти множества содержат коды с различными значениями r : $\{1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9, 9/10$ (9/10 — только для длинных кодов)}.

Предлагаемый декодер реализует итеративный алгоритм декодирования Message Passing (MP) [3], согласно ему, верные значения битов кодового слова определяются в результате многократного обмена сообщениями между вершинами графа Таннера. Всего выполняется n итераций (см. рис. 2, а). Каждая итерация алгоритма включает две фазы: в первой обновляются сообщения проверочных вершин на основе анализа сообщений кодовых вершин; во второй — сообщения кодовых вершин на основе анализа сообщений проверочных вершин.

Сообщения кодовых вершин вычисляются по формуле:

$$v_{n \rightarrow k_i} = u_n + \sum_{j \neq i} w_{j \rightarrow n}$$

где $v_{n \rightarrow k_i}$ — сообщение от n -й кодовой к k_i -й проверочной вершине; $w_{j \rightarrow n}$ — сообщение от j -й проверочной вершины к n -й кодовой; u_n — логарифмическое отношение правдоподобия (LLR) n -го бита, принятого из канала кодового слова.

Генерация сообщений проверочных вершин выполняется согласно формуле

$$w_{k \rightarrow n_i} = g(v_{n_1 \rightarrow k}, v_{n_2 \rightarrow k}, \dots, v_{n_{i-1} \rightarrow k}, v_{n_{i+1} \rightarrow k}, \dots, v_{n_{dc} \rightarrow k}),$$

где

$$g(a, b) = \text{sign}(a) \text{sign}(b) \{ \min(|a|, |b|) \} + LUT_g(a, b), \quad LUT_g(a, b) = \ln(1 + e^{-|a+b|}) - \ln(1 + e^{-|a-b|}).$$

Предлагаемый декодер LDPC-кодов обрабатывает сообщения параллельно группами (строками) с использованием техники двух проходов: в 1-м сообщения из памяти поступают в процессорные блоки, где рассчитываются и накапливаются промежуточные значения; во время 2-го одновременно рассчитываются и записываются в память встречные (относительно соответствующих ребер графа Таннера) выходные сообщения.

Организация обработки сообщений декодером представлена на рис. 2, б [4]: Z — степень распараллеливания процедуры обработки сообщений (число процессорных блоков, число столбцов памяти сообщений), для LDPC-кодов, используемых в стандарте DVB-S2,

$Z = 360$; q — параметр кода, $q=m/Z$; $u[i]$ — логарифмические отношения правдоподобия (LLR) битов принятого из канала кодового слова, $LDPC_out[i]$ — твердые решения относительно битов кодового слова.

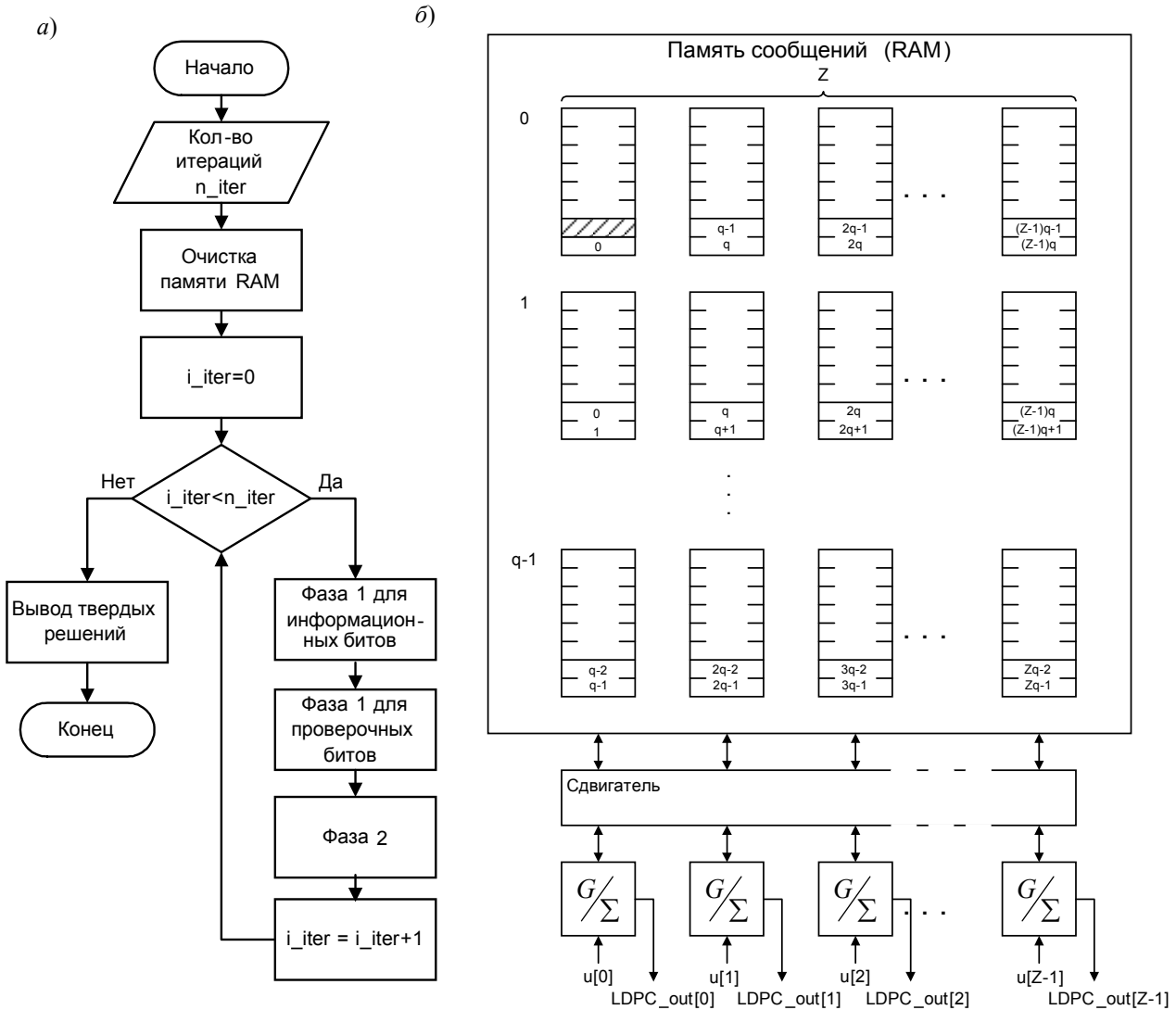


Рис. 2

Память сообщений содержит Zq блоков сообщений. В каждом блоке хранятся сообщения, относящиеся к одному проверочному уравнению из матрицы \mathbf{H} . После первой фазы итерации эти сообщения являются входными по отношению к проверочной вершине, после второй — выходными.

При обработке информационных битов в первой фазе группы сообщений, поступающих в линейку процессорных блоков, формируются на основе строк структуры описания кода. Адреса строк сообщений в памяти и значения сдвигов вычисляются предварительно и хранятся в ROM декодера.

При обработке проверочных битов строки сообщений сохраняются по адресам двух нижних ячеек блоков в памяти сообщений. Проверочные биты подаются на вход декодера совместно с перемежением q . Единичный сдвиг осуществляется только для первой строки последней группы данных.

Структурная схема декодера LDPC-кодов приведена на рис. 3 (Data Buffer — буфер приема входных значений LLR битов принятого из канала слова; Processing unit — модуль, содержащий Z процессорных блоков; Shifter1 и Shifter2 — сдвигатели; Sel — селектор;

Message RAM — память сообщений декодера; ROM AMR — ПЗУ, используемое для хранения адресов и сдвигов строк сообщений; Cnt AWDB — счетчик адресов записи буфера приема входных данных; Cnt gr — счетчик групп символов кодового слова; Rg BADB — регистр базового адреса буфера приема входных данных; Cnt AMRCN — счетчик адресов памяти сообщений, используемый при обработке сообщений проверочных вершин; Cnt AMRCB — счетчик адресов памяти сообщений, используемый при обработке проверочных битов; Cnt ROM AMR — счетчик адресов ROM AMR; Bit/Check node — признак, определяющий вид сообщений (кодowych вершин или проверочных), обрабатываемых в текущей фазе итерации; Inf/Check bit — признак, определяющий принадлежность LLR, используемых в фазе обработки сообщений кодowych вершин, к информационным или проверочным битам; LDPC_out — выход декодера, на который выдаются символы исправленного слова LDPC-кода).

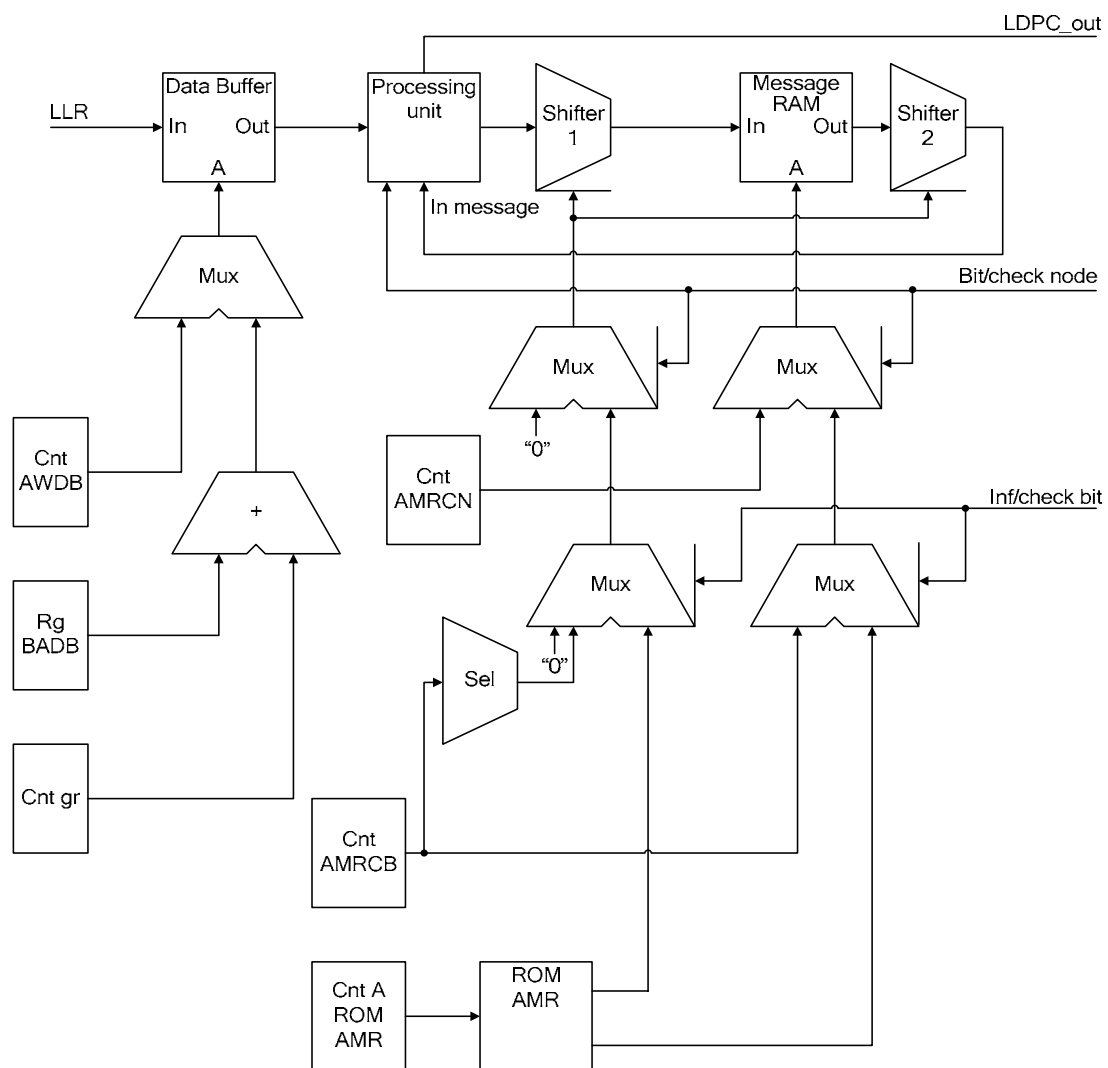


Рис. 3

LDPC-декодер выполняет итерации алгоритма декодирования следующим образом. В первой фазе активный признак Bit/Check node, подаваемый на вход модуля процессорных блоков, мультиплексора адресов RAM и мультиплексора сдвигов, указывает, что производится обработка сообщений кодowych вершин. Значения LLR битов принятого слова группами по Z значений передаются на обработку в модуль процессорных блоков Processing unit. Одновременно в Processing unit из памяти сообщений Message RAM, через сдвигатель Shifter2, подаются строки сообщений проверочных вершин. Каждая строка содержит Z сообщений.

Если в данной фазе активен признак Inf/Check bit (обрабатываются информационные биты), взятие адресов RAM для строк сообщений и величин сдвигов в сдвигателях Shifter1 и Shifter2 осуществляется из ROM AMR памяти декодера. Адреса ячеек данных в памяти ROM AMR определяются счетчиком Cnt ROM AMR.

Если признак Inf/Check bit не активен (обрабатываются проверочные биты), адреса RAM для строк сообщений определяются с помощью счетчика Cnt AMRCB. Сдвиг строк сообщений в этом случае не выполняется, за исключением одного адреса RAM, относящегося к первой строке последней группы LLR проверочных битов. В момент обработки этой строки, определяемый с помощью селектора, величина сдвига устанавливается на значении „1“.

После обработки каждой строки сообщений в первом проходе показания соответствующих счетчиков адресов увеличиваются на единицу.

После завершения обработки в первом проходе всех строк сообщений, относящихся к одной группе данных кодового слова, в модуле процессорных блоков Processing unit производится формирование сообщений кодовых вершин и запись их в память через Shifter1. При этом значения сдвигов и адреса RAM для записи определяются аналогично случаю первого прохода, однако счетчики адресов во втором проходе работают как вычитающие, и их показания уменьшаются на единицу после записи каждой новой строки сообщений.

Для каждой группы символов кодового слова на выходе модуля процессорных блоков Processing unit после завершения первого прохода генерируются твердые решения.

После обработки всех групп символов кодового слова декодер переключается во вторую фазу, в которой строки из Z сообщений кодовых вершин считываются из памяти сообщений Message RAM и подаются на вход модуля процессорных блоков Processing unit через Shifter2. Адреса строк сообщений в памяти определяются счетчиком Cnt AMRCN, который после считывания каждой строки увеличивается на единицу. Сдвиги строк сообщений на данной фазе не осуществляются.

После считывания и передачи в модуль процессорных блоков всех строк сообщений одной группы проверочных вершин начинается обратный проход, и формируются строки сообщений проверочных вершин. Затем строки сообщений записываются в память через Shifter1. Адреса RAM для записи строк сообщений также определяются счетчиком Cnt AMRCN, который в данном проходе является вычитающим.

После обработки строк сообщений всех проверочных вершин декодер переходит к выполнению первой фазы следующей итерации. Декодирование кодового слова заканчивается после выполнения заданного числа итераций.

Структурно-функциональная организация разработанного декодера LDPC-кодов характеризуется тем, что

- 1) обработка сообщений кодовых и проверочных вершин выполняется параллельно с использованием Z процессорных блоков, что обеспечивает высокое быстродействие декодера;
- 2) использование одного блока памяти для хранения сообщений как кодовых, так и проверочных вершин позволяет уменьшить объем встроенной памяти в декодере;
- 3) последовательная обработка процессорными блоками строк сообщений, относящихся к одной группе кодовых/проверочных вершин, с использованием двух проходов позволяет снизить число вентилях, необходимых для реализации декодера.

Для исследования эффективности исправления ошибок была разработана программная модель предлагаемого декодера LDPC-кодов. Была исследована исправляющая способность декодера в канале с гауссовым шумом (AWGN) и модуляцией QPSK.

На рис. 4 приведены графики зависимостей FER (Frame Error Ratio) на выходе LDPC-декодера от E_b/N_0 (отношение энергии сигнала на информационный бит к односторонней спектральной плотности шума) для нескольких LDPC-кодов, используемых в системе DVB-S2.

Обозначение кривой n соответствует длинным LDPC-кодам, s — коротким, id — зависимостям FER от E_b/N_0 для идеального декодера (без учета реализационных потерь) [5].

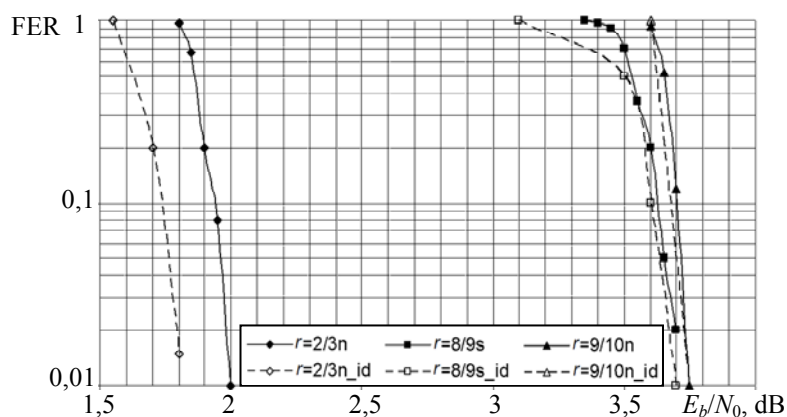


Рис. 4

Результаты проведенных исследований показали небольшое значение потери исправляющей способности, обусловленное неидеальной реализацией декодера (0,2 db для кода $2/3n$, менее 0,1 db — для коротких кодов).

Разработанный декодер LDPC-кодов для приемников цифрового телевидения второго поколения обеспечивает малые реализационные потери исправляющей способности и уменьшенную аппаратную сложность реализации.

Представленная работа поддержана грантом Федеральной целевой программы „Научные и научно-педагогические кадры инновационной России“ № 14.В37.21.0621.

СПИСОК ЛИТЕРАТУРЫ

1. DVB-S2. Стандарт. ETSI EN 302 307 V1.1.1 (2005-03) Digital Video Broadcasting (DVB).
2. Морелос-Сарагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. М.: Техносфера, 2005. 320 с.
3. Eroz M., Sun F.-W., Lee L.-N. DVB-S2 low density parity check codes with near Shannon limit performance // Intern. J. of Satellite Communications and Networking. 2004. Vol. 22, N 3. P. 269—279.
4. Егоров С. И., Чупахин А. Ю. Вычислительный алгоритм декодирования LDPC-кодов стандарта DVB-S2 // 7-я Междунар. науч.-техн. конф. „Опτικο-электронные приборы и устройства в системах распознавания образов, обработки изображений и символьной информации“. Курск: КГТУ, 2005. С. 137—139.
5. Valenti M. C., Cheng S., Iyer Seshadri R. Turbo and LDPC codes for digital video broadcasting // Chapter 12 of Turbo Code Applications: A Journey from a Paper to Realization. Springer, 2005.

Сведения об авторах

- Сергей Иванович Егоров** — д-р техн. наук, профессор; Юго-Западный государственный университет, кафедра вычислительной техники, Курск; E-mail: sie58@mail.ru
- Вадим Олегович Авдеев** — аспирант; Юго-Западный государственный университет, кафедра вычислительной техники, Курск; E-mail: v.o.avdeev@gmail.com
- Эдуард Игоревич Ватутин** — канд. техн. наук, доцент; Юго-Западный государственный университет, кафедра вычислительной техники, Курск; E-mail: evatutin@rambler.ru
- Владимир Славиевич Панищев** — канд. техн. наук, доцент; Юго-Западный государственный университет, кафедра вычислительной техники, Курск; E-mail: gskunk@rambler.ru

Рекомендована Юго-Западным государственным университетом

Поступила в редакцию 18.02.13 г.