Front. Optoelectron. China 2009, 2(4): 384–388
DOI 10.1007/s12200-009-0053-5

RESEARCH ARTICLE

Analysis and implementation of FEC in 10G-EPON

Guozhou JIANG (✉), Deming LIU

School of Optoelectronic Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

Abstract Forward error correction (FEC) may reduce the bit-error ratio (BER) and provide coding gain, so FEC is considered as a mandatory part in 10 Gbit/s Ethernet passive optical networks (10G-EPONs). First, the basic concepts of FEC and Reed-Solomon (RS) code are introduced in this paper. After a detailed description of the implementation of FEC in 10G-EPON, computed result shows that FEC based on $R_{RS}(255,223)$ can meet the need of 10G-EPON's high power budget. Therefore, FEC is the performance choice with high added value in 10G-EPON.

Keywords forward error correction (FEC), 10 Gbit/s Ethernet passive optical network (10G-EPON), Reed-Solomon (RS) code, bit-error ratio (BER), coding gain, optical line terminal (OLT), optical network unit (ONU)

1 Introduction

As the deployment of fiber to the home (FTTH) technologies in the access network accelerates, vendors and technology innovators are looking ahead to solve the bandwidth requirements of next-generation applications. These applications, such as high-definition Internet protocol television (IPTV) delivery and multimedia distribution systems, will demand far greater bandwidth than what current broadband access technologies provide. One attractive new option is the 10 Gbit/s Ethernet passive optical network (10G-EPON) technology. It offers a ten-fold leap in bandwidth to 10 Gbit/s in the broadband access network over fiber while providing core protocol compatibility with current 1G-EPON solutions.

EPON covers 20 km in distance, and the typical optical splitter ratios are 1∶32 and 1∶64. In addition, users and service providers believe that Ethernet passive optical network (EPON) is a high-performance, accurate and reliable system, so designers always adopt optical devices with good performance and low price to improve the optical power budget. Forward error correction (FEC) may reduce the bit-error ratio (BER) and provide coding gain, so FEC is considered as a mandatory part in 10G-EPON.

Received April 14, 2009; accepted April 28, 2009

E-mail: jgz_hust@163.com

2 Implementation of FEC in 10G-EPON

In the 10G-EPON draft specification, IEEE 802.3av, FEC seems to be considered as a mandatory part of the budget; how to implement FEC is important. One goal is to make everything integrally related to avoid fragmentation issues; other goals possess performance, flexibility and 1G-EPON compatibility.

2.1 Reed-Solomon code

FEC is a system of error control for data transmission, whereby the sender adds redundant data to its messages, also known as an error correction code. This allows the receiver to detect and correct errors.

One of the most widely used codes is the Reed-Solomon (RS) codes. RS codes are block-based error-correcting codes, and encoding and decoding are done on one block at a time. An RS code is denoted as $R_{RS}(n,k)$, where n is the length of the encoded block, k is the length of information block.

The RS encoding operates not over individual bits, but over m-bit symbols. The length of encoded block n is related to symbol size as $n = 2^m - 1$. Thus, if 8-bit symbols are used, n should be 255. The RS encoder takes a block of k information symbols and adds $n - k$ redundant symbols to it. The error-correcting capability of $R_{RS}(n,k)$ code is determined by the number of parity symbols $n - k$, and up to $(n - k)/2$ erroneous symbols per n-symbol block can be corrected.

To differentiate an FEC-encoded frame from a non-encoded frame, special frame markers are used. These frame markers need to be processed by the receiver before a frame can be delineated and the parity data can be

accessed. Therefore, the frame markers are not protected by the FEC. To reduce the probability of false marker detection or misdetection under high BER, the markers use a longer sequence of symbols. Table 1 shows the frame markers and their corresponding symbol sequence [1].

Figure 1 presents the structure of an FEC-coded frame. An FEC-coded frame starts with the /S_FEC/ sequence, between frame check sequence (FCS) and check code parity, T_FEC may be /T_FEC_O/ or /T_FEC_E/. But after check code parity, T_FEC is only to be /T_FEC_E/.

2.2 Stream-based versus frame-based FEC

A stream-based mechanism treats the Ethernet frames and idles between them as a stream of (uninterpreted) data symbols. The stream-based FEC is simpler to implement. Of course, stream-based encoding requires both the transmitting and the receiving devices to use this framing structure. A non-FEC-capable device will unavoidably become confused by added parity data and would not be able to recover any data. In EPON, this dependency translates to a situation such that if one optical network unit (ONU) needs to use FEC, all ONUs must use FEC, and conversely, if one ONU is unable to use FEC, none of the ONUs may use FEC.

A frame-based FEC method seeks to encode only the useful data (i.e., Ethernet frames) and to leave the gaps between the frames unprotected. In this method, a frame is divided into k-byte blocks, and $n-k$ bytes of parity data is added for each block. Depending on the frame length, the last block may be shorter than k bytes. Such a block is padded with zeros to the length of k, and the parity codes are calculated over the full-size block. However, the padding symbols are not transmitted. Similarly, the receiver reconstructs the shortened block to its full length by appending the necessary number of zeros, before applying the FEC decoder function to correct possible errors. In the frame-based method, the parity symbols generated for each block are grouped together and are appended at the end of a frame, leaving the frame itself unchanged. The fact that the entire Ethernet frame is left unchanged is the major advantage of the frame-based FEC encoding. It allows a non-FEC-capable device to receive an FEC-encoded frame, albeit without any error correction. Thus, an EPON can contain a combination of FEC-capable and FEC-incapable ONUs. Only the FEC-capable ONUs will take advantage of the added FEC protection. FEC-incapable ONUs will not see any coding gain, but nevertheless will be able to receive frames. Among the shortcomings of the frame-based FEC scheme, the main one is its variable overhead, which depends on a mix of packet sizes. Despite its higher complexity and higher overhead, the frame-based FEC was adopted as a baseline proposal for 10G-EPON.

2.3 Location of FEC in Ethernet stack

The FEC encoder is located in the extended physical coding sublayer. There was a dilemma on the placement of the FEC encoder relative to the existing physical coding sublayer (PCS) transmit function. On one hand, the FEC encoder substitutes the default frame delimiters (start-of-packet delimiter (SPD) and end-of-packet delimiter (EPD)) with /S_FEC/ and /T_FEC/, therefore, the FEC encoder should be placed after the PCS transmit state machine. On the other hand, the FEC encoder should operate on 64-bit data, not on 66-bit data (otherwise, overhead increases significantly), so it should be placed before the PCS transmit state machine which does 64 bit/66 bit encoding. Therefore, FEC should be applied at the lowest layer, below the 64 bit/66 bit sublayer, right before the physical medium attachment (PMA) sublayer, as shown in Fig. 2.

2.4 Encoding procedure

As a 66-bit coded frame enters the FEC encoder, it is passed to the 66 bit/64 bit decoder. The decoded frame, as a stream of 64-bit symbols, is shifted through a data buffer, and the parity symbols are calculated using the $R_{RS}(n,k)$ encoder. The unmodified frame passes through the selector and delimiter-detector (SDD) block. First, the SDD detects sequence /I/I/S/ or /T/R/I/, it replaces them with /S_FEC/ or /T_FEC_x/. When FEC is enabled, the interframe gap is increased to allow transmission of parity data, courtesy of the control multiplexer process. Following the transmission of /T_FEC_O/ or /T_FEC_E/, the SDD replaces the

Table 1 Frame delimiters for FEC-coded frames

notation	description	sequence
/S_FEC/	start of FEC-coded packet	/K28.5/D6.4/K28.5/D6.4/S/
/T_FEC_E/	end of FEC-coded packet with even alignment	/T/R/I/T/R/
/T_FEC_O/	end of FEC-coded packet with odd alignment	/T/R/R/I/T/R/

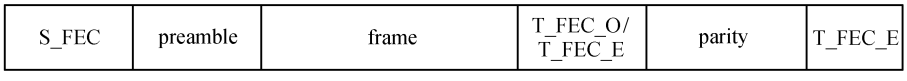

Fig. 1 FEC frame structure

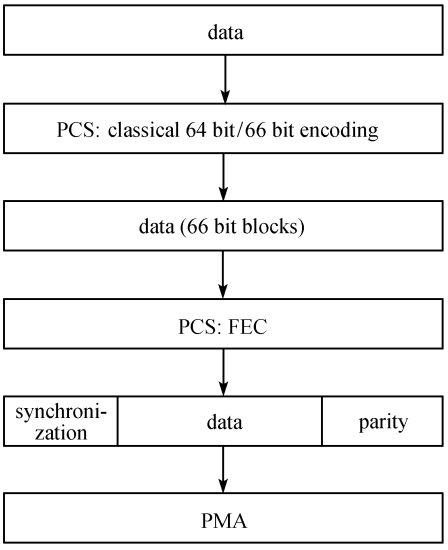

Fig. 2 Location of FEC in Ethernet stack

idle ordered sets in this extended interframe gap with the data from the parity buffer. When the parity buffer becomes empty, the SDD transmits /T_FEC_E/ and then transmits data (idles) emanating from the data buffer, until the /I/I/S/ sequence is detected. Finally, the 64-bit data out of SDD passes through the 64 bit/66 bit encoder again, and the complete FEC-coded frame is passed to the data detector which is responsible for turning the laser on and off. Figure 3 shows the block diagram of the FEC encoding function.

2.5 Decoding procedure

Since FEC and non-FEC ONUs may be combined in the same EPON, the FEC decoding procedure should not only process and correct errors in FEC-coded frames, but also be able to transparently pass any non-FEC-coded frame to the regular PCS receive process.

As the case with FEC encoder, the FEC decoder has a sixteen bit interface (XSBI) on both ends, as shown in Fig. 4. Thus, implementations that do not require error correction capabilities may exclude the FEC decoder without affecting the adjacent functions. The FEC parity data are appended at the end of the frame. Therefore, an entire frame should be buffered before the parity data can be accessed. The buffering delay is determined by the maximum frame size plus the necessary time to perform error correction. For normal multi-point control protocol (MPCP) operation, the delay between the control multiplexer and control parser should be constant. The bypass buffer provides a matching delay for non-FEC-coded

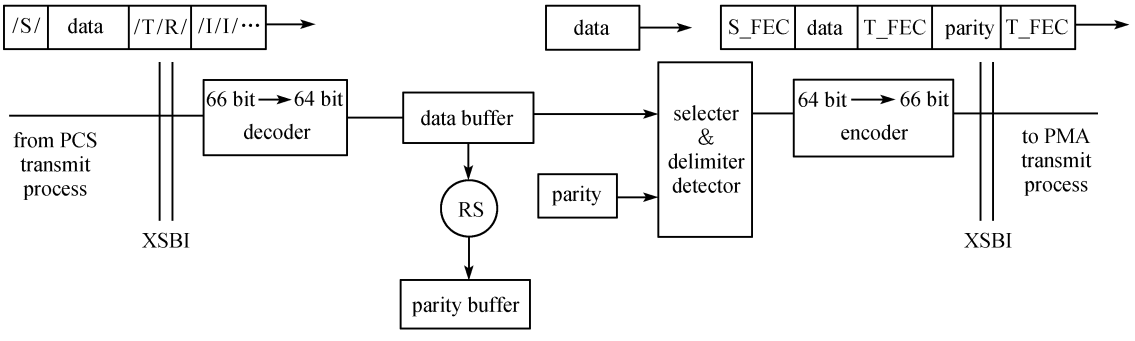

Fig. 3 FEC encoder block diagram (XSBI: sixteen bit interface)

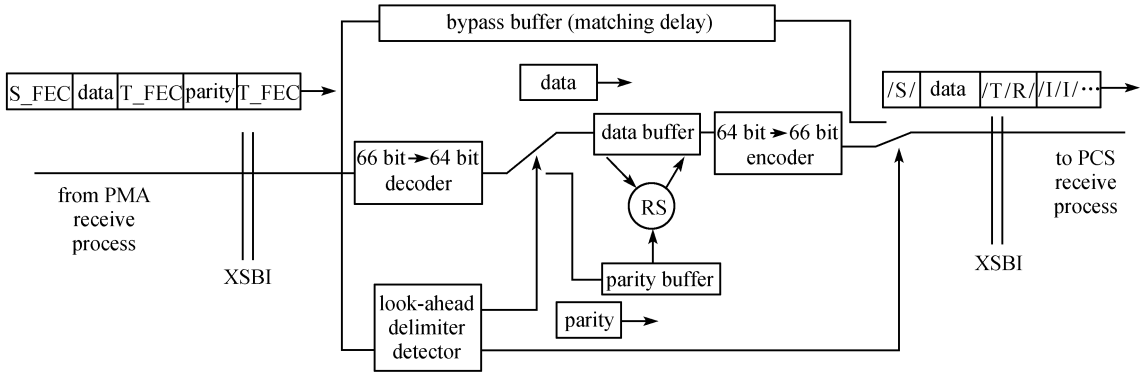

Fig. 4 FEC decoder block diagram

frames. If an arriving frame is FEC-coded, it is passed to the 66 bit/64 bit decoder. The decoded frame is stored in the data buffer until its parity data arrive. When the look-ahead delimiter detector matches the first /T_FEC_x/, it directs the remaining portion of the incoming data, up to the second /T_FEC_E/, to the parity buffer. Once the first 16 bytes of the parity data is received, the error correction of the first block can begin. The corrected 16-bit data are encoded into 66-bit code-words again and passed to the existing PCS receive process. The FEC decoder replaces the /S_FEC/ delimiter with an /I/I/S/ sequence. The first /T_FEC_x/ delimiter is replaced with EPD followed by /I/I/. Finally, the parity data and the second terminating delimiter /T_FEC_E/ are replaced by idle ordered sets. Thus, to the regular PCS receive state machine, the received frame looks like a regular Ethernet frame with an extended run of idles (interframe gap) at the end [2].

2.6 RS code application and results

The generator polynomial of the $R_{RS}(n,k)$ code is given as

$$G(z)=\prod_{i=0}^{2r-1}(d_i-\alpha^i),$$

where $r=3,4,\ldots$, is the errors correcting capability of the RS code, α is a root of the binary primitive polynomial $x_8+x_4+x_2+1$. A data byte $(d_7,d_6,\ldots,d_1,d_0)$ is identified with the element $d_7\alpha_7+d_6\alpha_6+\cdots+d_1\alpha_1+d_0$ in $G_{GF}(256)$, which is also called the Galois field (GF), in honor of the founder of finite field theory. Here, we suppose $n=255$.

A criterion for the evaluation of the intrinsic correcting performance of the $R_{RS}(n,k)$ code is the theoretical relationship between the line BER before FEC function $B_{BER(in)}$ and the line correction BER after FEC function $B_{BER(out)}$. For the RS codes, this criterion can be mathematically computed with the assumptions that errors occur independently from each other and that the decoder never fails (probability of incorrect decoding equal to zero) [3]:

$$P_{out}=\sum_{i=r+1}^{n}\frac{i}{n}\frac{n!}{i!(n-i)!}P_{in}^{i}(1-P_{in})^{n-i},$$

$$B_{BER(in)}=1-(1-P_{in})^{1/8},$$

$$B_{BER(out)}=1-(1-P_{out})^{1/8},$$

where P_{out} is the output symbol error rate; P_{in} is the input symbol error rate; n is codeword length (255).

Theoretical output BER results versus input BER are shown in Fig. 5, from very high input $B_{BER(in)}$, we can easily get a satisfactory output $B_{BER(out)}$ either using high error correcting capability (i.e., $r=8,16$) or low error correcting capability (i.e., $r=3,4$), but FEC overhead will increase as r increases.

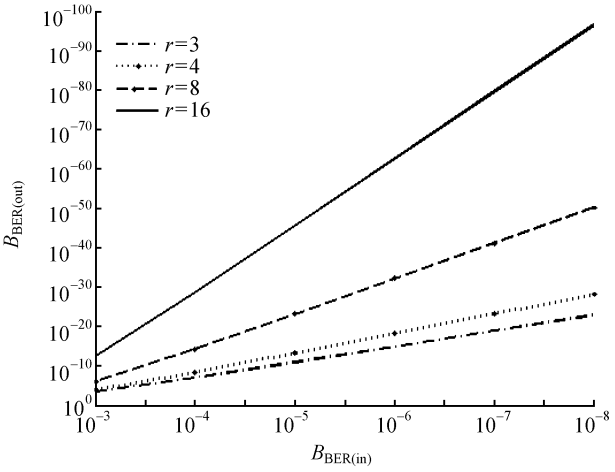

Fig. 5 Theoretical performance of RS code

The correcting capability of FEC is usually measured coding gain, which can be defined as

$$F_{FEC\_CG}=20\log Q_{out}-20\log Q_{in},$$

$$B_{BER}=\frac{1}{2}\mathrm{erfc}\left(\frac{Q}{\sqrt{2}}\right).$$

According to theoretical predictions in Fig. 5, we can easily get the result of Q. For given correcting capability $r=3,4,8,16$, a link operating at $Q\sim3.72$ dB would have a $B_{BER(in)}\sim10^{-4}$. After correction, the corresponding output BER would be 1.19×10^{-7}, 5.90×10^{-9}, 5.40×10^{-15}, 2.07×10^{-29}, and the corresponding Q would be 5.17 dB, 5.70 dB, 7.73 dB, 11.20 dB. The code gain benefits are 2.86 dB, 3.70 dB, 6.35 dB, 9.57 dB.

3 Conclusions

In 1G-EPON, high power budget classes are required to support point to multi-point media 24 dB channel insertion loss (ChIL). Employing FEC based on $R_{RS}(255,239)$ [4] code may meet it, but in 10G-EPON, high power budget classes are required to support point to multi-point media 29 dB ChIL. Therefore, the FEC based on $R_{RS}(255,223)$ code is employed, which may meet the needs of the physical layer medium, and also reduce the cost of the optical device effectively. Of course, the gain provided by FEC also can be used to increase the distance between the optical line terminal (OLT) and ONUs, or to increase the split ratio of EPON, or simply to improve the reliability of the digital channel.

Acknowledgements This paper was supported by the High Technology Research and Development Program of China (No. 2007AA01Z229).

References

1. Lee H. A high-speed low-complexity Reed-Solomon decoder for optical communications. IEEE Transactions on Circuits and Systems, 2005, 52(8): 461–465
2. Kramer G. Ethernet Passive Optical Networks (in Chinese, trans. Chen Xue). Beijing: Beijing University of Posts and Telecommunications Press, 2007, 121–123(Original work published 2005)
3. Schmitt A. Improving optical networks with forward error correction. Electronic Engineering (London), 2000, 72(878): 87–88
4. IEEE 802.3ah Ethernet in the First Mile Task Force. IEEE standard 802.3ah TM. 2004