Bit-Stuffing Rate in the High-Level Data Link Control (HDLC) Protocol

Philip Branch, Sebastian Zander
Centre for Advanced Internet Architectures. Technical Report 081121A
Swinburne University of Technology
Melbourne, Australia
{pbranch, szander} @swin.edu.au

***Abstract*-This technical report presents an analysis and simulation of the frequency with which bit stuffing occurs in the High-Level Data Link Control (HDLC) protocol.**

***Keywords*- HDLC, Bit stuffing**

I. Introduction

High-Level Data Link Control (HDLC) is a widely used, bit-oriented synchronous data link layer protocol. This paper answers the following question. In an HDLC frame of arbitrary length where the sequence of data bits is entirely random, what is the frequency of bit stuffing?

In HDLC, the start and end of each frame is identified by the bit sequence 01111110, referred to as the Sync character. Sync characters are used in bit-oriented synchronous protocols to allow the beginning and end of each frame to be identified and to allow the receiver to synchronize itself with the transmitter.

However, in bit oriented protocols, unlike byte oriented protocols, there are no restrictions as to the actual bit stream transmitted. Consequently, the bit sequence 01111110 may occur in the data stream. If data consisting of this sequence were to be transmitted, the receiver would incorrectly interpret it as the end of the frame. To avoid this, 'bit stuffing' is used. When the transmitter encounters five consecutive 1-bits, it inserts a 0-bit before transmission. The receiver, upon encountering five consecutive 1-bits removes the following 0-bit [1].

This work was carried out as part of an investigation into bit rates of covert channels based on bit stuffing errors [2, 3]. To determine capacity of such channels an accurate measure of the overhead of HDLC was needed.

In the next section we present an analysis and simulation of the rate at which bit stuffing occurs.

II. Frequency of bit stuffing in HDLC

A. *Analysis*

Consider the data field X of an HDLC frame of arbitrary length N where the bit sequence $X = \{x_1, x_2, x_3, \ldots, x_N\}$ is uniformly random and where a 0-bit is inserted after n consecutive 1-bits. If we segment the sequence X into subsequences of n consecutive bits, there will be $N - n + 1$ such subsequences. The probability that any such subsequence will be all 1-bits is 2^{-n}.

However, the act of bit stuffing affects the number of stuffable subsequences. This is because stuffable subsequences are highly correlated. If the subsequence $x_k, x_{k+1}, x_{k+2}, \ldots, x_{k+n-1}$ is all 1-bits the probability that the subsequence $x_{k+1}, x_{k+2}, \ldots, x_{k+n-1}, x_{k+n}$ is also all 1-bits is considerably greater than 2^{-n}. In fact, since the sequence is uniformly random, then x_{k+n} will be a 1-bit with a probability of 0.5. The probability that the next subsequence is also all 1-bits is 0.25 and so on. Consequently, whenever bit stuffing occurs, an additional number of subsequences of n 1-bits may be lost. The expected number of subsequences lost will be:

$$L = \sum_{i=1}^{n-1} \frac{1}{2^i}. \qquad (1)$$

So bit stuffing reduces the number of stuffable subsequences by a factor of $1 + L$. That is, the rate at which bits will be stuffed is:

$$R = \frac{2^{-n}}{(1+L)}. \qquad (2)$$

Note that this is independent of the length of the sequence N.

We can make the useful observation that L has an upper bound of 1 approached as n becomes large. That is, if we denote the bit stuffing rate for large n by R_∞, then

$$\begin{aligned} R_\infty &= \frac{2^{-n}}{(1+L)} \\ &= \frac{2^{-n}}{\left(1+\sum_{i=1}^{n-1} \frac{1}{2^i}\right)} \\ &\cong 2^{-n-1}. \qquad (3) \end{aligned}$$

We now illustrate this with an example. In HDLC n is 5 [1]. If the subsequence $x_{k+1}, x_{k+2}, x_{k+3}, x_{k+4}, x_{k+5}$ is all 1-bits the probability that the subsequence $x_{k+2}, x_{k+3}, x_{k+4}, x_{k+5}, x_{k+6}$, is also all 1-bits is 0.5. However, after bit stuffing, this subsequence will no longer be all 1-bits. By calculating the expectation of the number of all 1-bit subsequences lost (L) we see that:

$$L = 0.5 + 0.25 + 0.125 + 0.0625$$
$$= 0.937.$$

Without stuffing we would expect subsequences consisting of all 1-bits to occur at the rate of 2^{-5} = 0.0313. However, the action of bit stuffing reduces the number of stuffable subsequences. Every time we stuff a subsequence we lose (on average) 0.937 additional stuffable subsequences. Consequently the rate at which we bit stuff a random subsequence will be

$$\frac{2^{-5}}{(1.937)} = 0.0161.$$

We now test this and other cases with simulation.

B. *Simulation code*

Follows is the simulation code written using the open source mathematical package Scilab [4]. It tests our analysis for n = 5. Similar code has been written for other scenarios which are reported in Section II.C.

```
%simulates bit stuffing for a large
%number of frames with a random length

totstuff = 0;
totsequences = 0;
for reps = 1:100 %number of frames
   N = floor(4000*rand) + 1 ;
     %number of bytes in this frame.
      %Max 4000. Average 2000. Min 1.
   nbits = N*8; %number of bits
   %initialise bit sequence
   bit = zeros(1, nbits);
   for i = 1:nbits
    bit(i) = 1;
    if rand > 0.5 bit(i) = 0;
    end
  end

  %do the bit stuffing
  i = 5;
  seqcount = 0;
  while i < length(bit)
    if (bit(i) == 1) && (bit(i-1) == 1) &&
       (bit(i-2) == 1) && (bit(i-3) == 1)
        && (bit(i-4) == 1)
      %bitstuff
      firstseq = bit(1:i);
      secondseq = bit(i:length(bit));
      bit = [firstseq [2] secondseq];
    end
    i = i+1;
    seqcount = seqcount + 1;
  end

  %how many stuff bits?
  count = 0;
  for i = 1:length(bit)
   if bit(i) == 2
    count = count+1;
   end
  end
  totsequences = totsequences + seqcount;
  totstuff = totstuff + count;
end

totstuff / totsequences
```

C. *Results*

Table 1 shows the results of simulating the bit stuffing rate. Clearly it strongly supports our analysis. We also note that for n = 8, the upper bound approximation of the bit stuffing rate of 2^{-n-1} is correct to four decimal places.

Table 1. Predicted and observed rate of bit stuffing

Stuffing length	Predicted	Simulated
2	0.1667	0.1667
5	0.0161	0.0161
7	0.0039	0.0039
8	0.0020	0.0020

III. Conclusion

In this paper we have shown how to model the frequency of bit stuffing of a 0-bit after n consecutive 1-bits. For the case where bit stuffing occurs after five consecutive 1-bits the rate is 0.0161. The rate is independent of the length of the frame and, as the stuffing length increases, trends towards 2^{-n-1}.

This work was carried out as part of an investigation into bit rates of covert channels based on bit stuffing errors and should be of benefit to it.

References

[1] W Stallings, "ISDN and Broadband ISDN with Frame Relay and ATM," 3rd edition, Prentice-Hall, New Jersey, 1995

[2] S. Zander, G. Armitage, P. Branch, "A Survey of Covert Channels and Countermeasures in Computer Network Protocols," IEEE Communications Surveys & Tutorials, vol.9, no.3, July 2007, pp.44-57.

[3] S. Zander, G. Armitage, "CCHEF – Covert Channels Evaluation Framework Design and Implementation," CAIA Technical Report 080530A, May 2008

[4] Scilab, "The open source platform for numerical computation," http://www.scilab.org/ accessed 19 Nov. 08