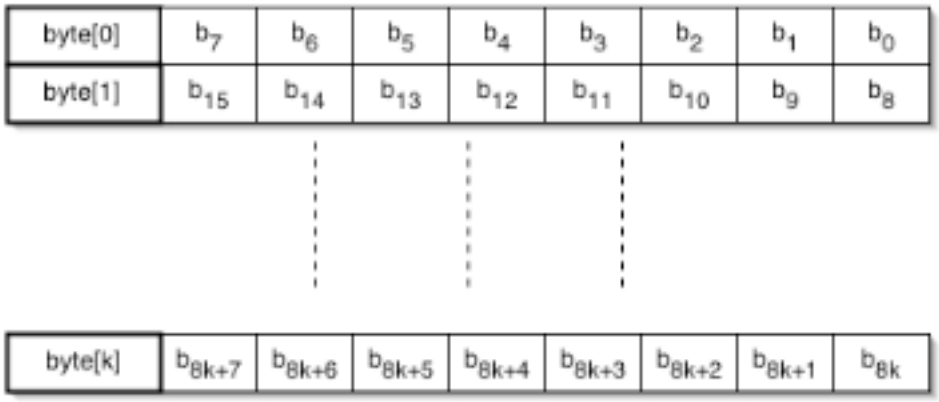
This note addresses the issues raised by comment #308 in Draft 1.1. The issue deals with the error detection properties of crc-32 when the bit transmission order within a byte is different form the ordering of bits used in crc-32 calculation. The three specific issues addressed are:

1. Single bit error detection characteristics
2. Burst error protection
3. Effect of scrambling polynomial x^43+1

Illustration of Problem Statement

The (k+1) byte packet can be denoted by byte[0], byte[1], ..., byte[k]. Bits within the packet bytes are illustrated below.



The CRC calculation (in a canonical sense, actual implementations process groups of bits however the end result is equivalent to the serial case) for bytes byte[0] through byte[k] is generated by feeding the linear feedback register the sequence of bits b_0, b_1, ..., b_7, ..., b_{8k}, ..., b_{8k+7}.

When the packet is transmitted over Sonet the transmission order of bits is in sequence b_7, ..., b_0, ..., b_{8k+7}, ..., b_{8k}.

When errors occur on the fiber the CRC calculation logic at the receiver will see bit reversed (within bytes) error patterns. The following sections investigate if there are any ramifications on the error detection behavior of crc-32 (as raised by comment #308).

Basics

It is well known that crc generation and checking is a linear process. Therefore, without any loss of generality we need only consider error patterns (represented by bits streams or error polynomials) and can assume the original packet data and the initial crc to be all zero.

1. Single Bit Error Detection Property

Since bit reversal is a permutation operation if k-bit errors occur in the transmission media they will be reflected as k-bit errors (although in different locations defined by the bit reversal operation) to the CRC calculation logic at the receiver. Specifically, all single bit errors are detected. All double bit errors are detected for packet sizes less than or equal to 2^32-1 bits (this is several orders of magnitude bigger than jumbo frames). This fact follows from the knowledge that crc-32 is a primitive polynomial. In general, if k-bit errors are equally likely across the packet length then the crc detection probability is identical with or without bit reversal in the transmission order.

2. Burst Error Detection Probability

It is a well known fact that monic degree-32 polynomial can detect burst errors up to 32 bits. A length *L burst error* spans up to *L* contiguous bits potential bit locations.

For comparative analysis we need only analyze burst errors up to length 32 in the transmission media. Because of bit reversal within bytes, burst errors of length up to 32 in the transmission media may not appear as of same length to the CRC calculation logic at the receiver. Figure 1. illustrates this.

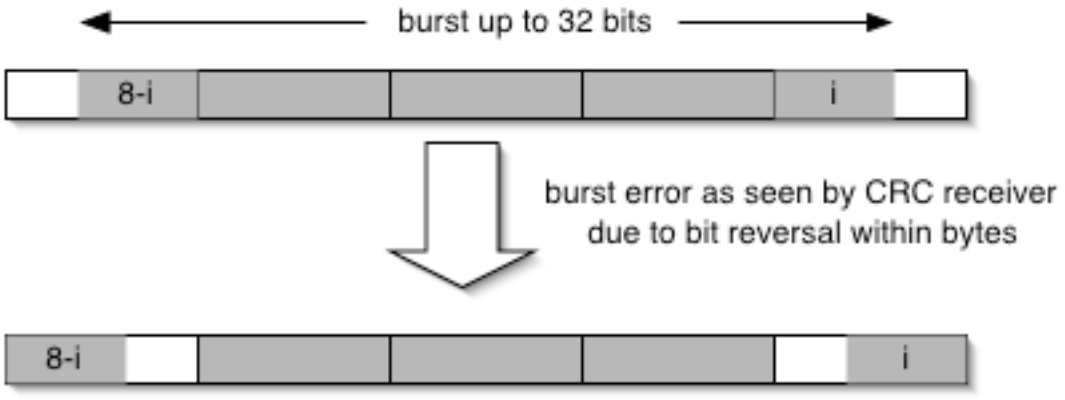


Figure 1. Rearrangement of Burst Error Bits Due to Bit Reversal within Bytes

There are eight different cases of interest. If the burst error in the transmission media is byte aligned (i.e, i=0) then at the receiver side the crc-32 polynomial will also see it as a burst error of length 32. So for i=0, all byte aligned 32-bit burst errors are detected. The other

seven cases of interest are for i=1,...,7.

We can do an exhaustive simulation to determine if all of some of these misaligned (i=1,...,7) burst errors are detected by the crc-32 polynomial. My estimate is that this simulation will take several days as we have to do iterations of order 7×2^{32}.
For now, we can provide an upper bound on the probability of error detection. The upper bound is based on simple principle of decomposition and linear super-position.

i	Probability of Undetected Burst Error (Less Than or Equal To)
0	All burst errors are detected. Probability of undetected burst error in this case is 0.
1	Let us decompose the burst error seen by the crc-32 polynomial into two parts. First part within the byte segment containing "i"; and the second part of the other 4-byte segment containing three contiguous bytes and the byte segment denoted by "8-i". Let us now seperately consider the potential crc values generated by these two parts. In the first part, there are 2^i-1 potential errors and each of these errors will produce 2^i-1 unique values of crc. Likewise, in the second part there are $2^{24+8-i}=2^{32-i}$ (including zero) unique values of crc. The unique values of crcs follow from two facts, 1) crc-32 polynomial is primitive and 2) each of the two parts is less than or equal to 32 bits. For the burst error to be not detected the crc values produced by both parts must be idtentical. This may or may not be possible and can be determined by simulation. If it is possible, then for every crc produced in part one there can be only one out of 2^{32-i} unique crc values in part two that can match. So the probability of a perfect match (and therefore undetected error) for any burst error cannot exceed $1/2^{32-i}$. Therefore in this case (i=1), the probability of undetected error is less than or equal to $1/2^{31}$.
2	$1/2^{30}$
3	$1/2^{29}$
4	$1/2^{28}$
5	$1/2^{27}$
6	$1/2^{26}$
7	$1/2^{25}$

Assuming all alignements of burst errors are equally likely then the probability of undetected burst is less tha or equal to $1/8 \times (1/2^{31} + ... + 1/2^{25}) \sim 2^{-27}$

Therefore the worst case upper bound on undetected burst error is 1 in 2^{27}. The actual probability may be much less than this (determined through simulations). If the working group is interested then I can start the exhaustive simulation.
However, I strongly feel that even this upper bound value is small enough to warrant any change in the PHY reconciliation layer to do bit reversal. This will have an impact on implementation using exisiting Sonet framers.

3. Influence of Scrambling Polynomial $x^{43}+1$

If the bit errors on the transmission media are represented by polynomial E(x) then at the receiver (after unscrambling) the error polynomial seen by the crc-32 logic will be equivalent to $(x^{43}+1)E(x)$. Undetected bit errors by crc-32 are those that would cause the error polynomial $(x^{43}+1)E(x)$ to be a multiple of the crc-32 polynomial. The factorization of $(x^{43}+1)$ [using Maple software) is as follows.

(x+1)*(1+x+x^6+x^5+x^4+x^3+x^2+x^19+x^17+x^16+x^15+x^14+x^13+x^12+x^11+x^10+x^9+x^8+x^7+x^18+x^42+x^41+x^40+x^39+x^38+x^37+x^36+x^35+x^34+x^33+x^32+x^31+x^30+x^29+x^28+x^27+x^26+x^25+x^24+x^23+x^22+x^21+x^20)

Since crc-32 is primitive polynomail it has only one factor (as shown below).

x^32+x^26+x^23+x^22+x^16+x^12+x^11+x^10+x^8+x^7+x^5+x^4+x^2+x+1

It is clear from the factorization of $(x^{43}+1)$ that it is relatively prime (no common multiple) to crc-32. Therefore for the error polynomial to be undetectable by crc-32, E(x) has to be a multiple of crc-32. This is exactly the same case if there were no scrambling.

Therefore, scrambling with $(x^{43}+1)$ has no impact on the error detection capability of crc-32.