

To: T10 Technical Committee
From: Luben Tuikov, Vitesse Semiconductor (ltuikov@vitesse.com)
Date: 20 March 2007
Subject: 07-130r2 SAS-2 CRC fixes

Revision history

Revision 0 (12 March 2007) First revision

Revision 1 (19 March 2007) Second revision

1) Table 112, row defining $R'(x)$:

a) Define $R'(x)$ explicitly.

b) Use $M(x)$ instead of $M'(x)$, as the latter is a designation of the former, which is what is received.

Revision 2 (20 March 2007) Third revision

Address editor's comments.

Related documents

The iSCSI CRC32C Digest and the Simultaneous Multiply and Divide Algorithm - Tuikov&Cavanna, Jan 30, 2002, although using a different generator polynomial, the paper describes the algebra in great detail.

Overview

Fix the math.

Convert to Frame Maker's equations format.

Suggested changes

7.5 CRC

7.5.1 CRC overview

All frames include cyclic redundancy check (CRC) values to help detect transmission errors.

Frames transmitted in an STP connection shall include a CRC as defined by SATA (see ATA/ATAPI-7 V3). Address frames, SSP frames, and SMP frames shall include a CRC as defined by this standard.

Annex D contains information on CRC generation/checker implementation.

Table 112 defines notation used in the following text describing the CRC calculation.

Table 112 — CRC notation and definitions

Notation	Definition
$T(i)$	A transformation over the non-negative integers, $T(i) = i + 7 - 2(i \bmod 8), i \geq 0, i \in \mathbb{Z}^+$
$F(x)$	A polynomial representing the frame contents which are covered by the CRC. Let the number of bits in the frame be k and the frame bits described by b_i , where the bit index i , only denotes that bit b_i is more significant than bit b_{i+1} , then, $F(x) = b_0x^{k-1} + b_1x^{k-2} + \dots + b_{k-2}x + b_{k-1}$
$F_t(x)$	A polynomial representing the frame contents, but bit positions of each byte are transposed. That is bit index 7 is now bit index 0, bit index 6 is now bit index 1, etc. Defined as, $F_t(x) = b_{T(0)}x^{k-1} + b_{T(1)}x^{k-2} + \dots + b_{T(k-2)}x + b_{T(k-1)}$
$L(x)$	The identity polynomial of degree 31. A polynomial with all of the coefficients set to one, $L(x) = x^{31} + x^{30} + \dots + x + 1$ (i.e., $L(x) = \text{FFFF_FFFFh}$).
$G(x)$	The CRC generator polynomial, the divisor polynomial, $G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$ (i.e., $G(x) = \text{1_04C1_1DB7h}$).
$R(x)$	A remainder polynomial, always of degree less than 32.
$R_t(x)$	A remainder polynomial, but bit positions of each byte are transposed. That is, the bit index to each term of this polynomial is $T(i)$, instead of i .
$Q(x)$	A quotient polynomial.
$Q'(x)$	A quotient polynomial.
$M(x)$	A polynomial of degree $k+31$ representing the frame followed by the CRC, as presented to the 8b10b encoder for transmission.
$M'(x)$	A polynomial representing the received frame followed by the received CRC, after it has been 8b10b decoded. It is of degree $k+31$, and equal to $M(x)$, for an error free reception.
$R'(x)$	The result of finding the remainder of an error free reception of $M(x)$. The remainder of $\frac{x^{32}L(x)}{G(x)}$, a unique constant polynomial, $R'(x) = x^{31} + x^{30} + x^{26} + x^{25} + x^{24} + x^{18} + x^{15} + x^{14} + x^{12} + x^{11} + x^{10} + x^8 + x^6 + x^5 + x^4 + x^3 + x + 1$ (i.e., $R'(x) = \text{C704_DD7Bh}$). Note that $R'(x)$ transposed and inverted is 1CDF_4421h , (i.e. $R'_t(x) + L(x) = \text{1CDF_4421h}$).

7.5.2 CRC generation

Arithmetic is modulo 2. The CRC, $R(x)$, of a frame is generated as follows,

$$x^kL(x) + x^{32}F_t(x) = Q(x)G(x) + R(x).$$

That is, the frame is transposed, then the first 32 bits of the transposed frame are inverted, then 32 bits of 0 are appended to the end of the transposed frame, then this result is divided by the generator polynomial to find the remainder, $R(x)$.

Note that, for the purposes of CRC generation, inverting the first 32 bits of a frame (i.e. seeding the CRC remainder register with FFFF_FFFFh) is equivalent to prepending the constant 62F5_2692h to an untransposed frame and seeding the CRC register with 0000_0000h .

The sender shall present to the 8b10b encoder,

$$M(x) = x^{32}F(x) + L(x) + R_t(x).$$

That is, the inverted transposed remainder is appended to the end of the frame, then this result is presented to the 8b10b encoder for transmission.

Figure 151, shows the CRC process for an address frame, an SSP frame and SMP frame.

Figure 151 — Address frame, SSP frame, and SMP frame CRC bit ordering

Dwords in STP frames are little-endian and feed into the STP CRC generator without swapping bits within each byte and inverting the output like the SAS CRC generator. Figure 152 shows the STP CRC bit ordering.

Figure 152 — STP frame CRC bit ordering

Since STP is little-endian, the first byte of a dword is in bits 7:0 rather than 31:24 as in SSP and SMP. Thus, the first byte contains the least-significant bit. In SSP and SMP, the first byte contains the most-significant bit.

See 7.7 for details on how the CRC generator fits into the dword flow along with the scrambler.

7.5.3 CRC checking

CRC of a received frame is generated by the receiver in the same manner that it is generated by the sender. A received frame which has not incurred any CRC detectable errors during transmission, should generate a remainder equal to $R'(x)$.

Denote a received frame by $M'(x)$. If there were no transmission errors the received frame would equal to $M(x)$ and of degree $k+32$,

$$M'(x) = x^{32}F(x) + L(x) + R_t(x).$$

The CRC, $R'(x)$, is derived as follows,

$$\begin{aligned} x^{k+32}L(x) + x^{32}(x^{32}F_t(x) + L(x) + R(x)) &= \dots \\ &= x^{32}Q(x)G(x) + x^{32}L(x). \end{aligned}$$

That is, the received frame, $M'(x)$, is transposed, then the first 32 bits of the received transposed frame are inverted, then 32 bits of 0 are appended to the end of the received transposed frame, then this result is divided by the generator polynomial to find the remainder.

But $G(x)$ divides $x^{32}L(x)$,

$$x^{32}L(x) = Q'(x)G(x) + R'(x).$$

In the above equation, $L(x)$ and $G(x)$ are known and constant, thus $R'(x)$ is known and constant. It is this $R'(x)$ which the receiver should get after calculating the CRC of a received frame.

From the previous two results,

$$x^{k+32}L(x) + x^{32}(x^{32}F_t(x) + L(x) + R(x)) = (x^{32}Q(x) + Q'(x))G(x) + R'(x).$$

$R'(x)$ is then transposed and inverted, in the same manner as is done by the sender, to obtain,

$$R_t'(x) + L(x) = 1CDF_4421h.$$

See 7.7 for details on where the CRC checker fits into the dword flow along with the descrambler.