BACKGROUND

Our state machine is looking for two FEC codewords to lock, the FEC codewords has the following pattern,

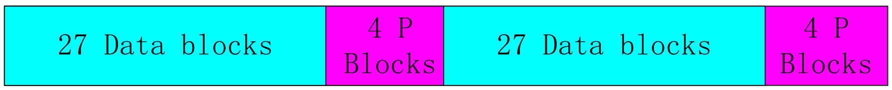

Figure 1. FEC codewords structure to Lock State Machine in 10G EPON

From the perspective of locking performance, the machine can be considered to have four states, and the following transitions and their associated probabilities, as shown below:

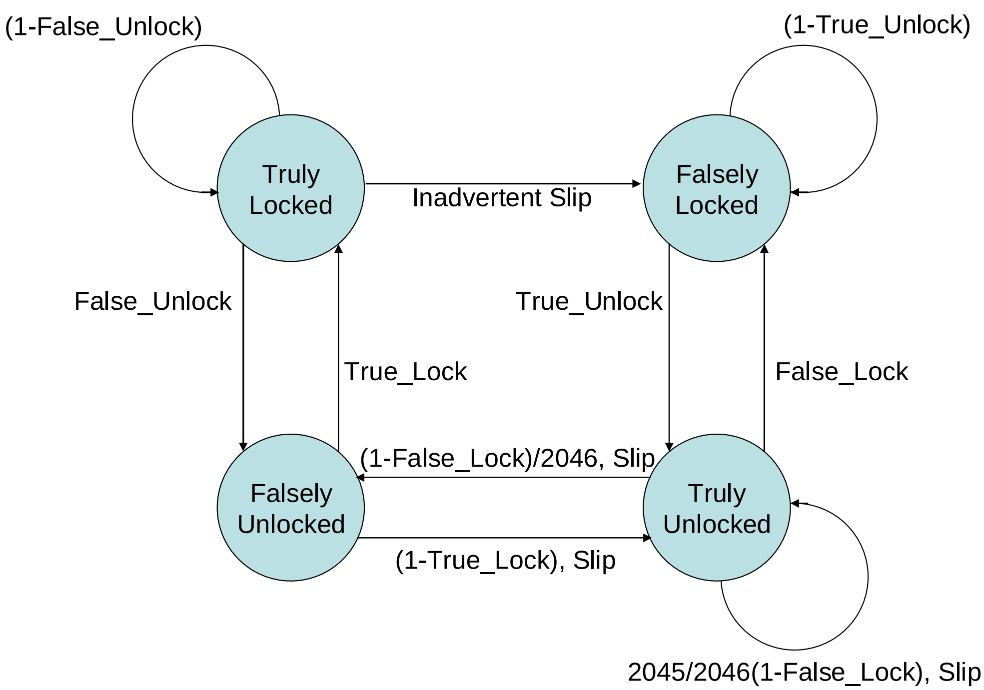

Figure 2 State transition diagram of the Lock State Machine

Now assuming the bit error probability is p then the bit correctability is $1-p$.

Let's consider the sync headers of the Data blocks,
00 – Invalid
01 – Valid
10 – Valid
11 – Invalid
If a true valid sync headers of a data block is 01 then,

Deleted: 2008-03-14

00 – Invalid (one bit error & one bit correct - p(1-p))
01 – Valid (two bits correct – (1-p)^2)
10 – Valid (two bits error – p^2)
11 – Invalid (one bit error & one bit correct - p(1-p))

Let's consider the sync headers of the parity blocks,
If a valid sync headers of a parity block is 00 then,
00 – Valid (two bits correct – (1-p)^2)
01 – Invalid (one bit error & one bit correct - p(1-p))
10 – Invalid (one bit error & one bit correct - p(1-p))
11 – Invalid (two bits error – p^2)

If a valid sync headers of a parity block is 11 then,
00 – Invalid (two bits error – p^2)
01 – Invalid (one bit error & one bit correct - p(1-p))
10 – Invalid (one bit error & one bit correct - p(1-p))
11 – Valid (two bits correct – (1-p)^2)

The following results are the four MTTs as follows, where p =10e-3

MTT-true-lock $\approx$ *18.6037 us(computer simulation)*
MTT-false-lock = 8.4e+011s
MTT-true-unlock = 0.4092 us
MTT-false-unlock_16 = 2.528e22 s, longer than life of universe. (after 16 errors)

Deleted: 6

MTT-true-lock

For true-lock to happen, it requires 62 continuous valid sync headers, e.g. the sync headers of data block is either 01 or 10 and the sync headers of parity blocks is 00 11 11 00.

We have 54 Data blocks, assuming i sync headers having 2 bits error and (54-i) sync headers having 2 bits correct then the probability of 54 correct sync headers of data blocks is,

$$\sum_{i=0}^{54} C_{54}^{i}(p^2)^i(1-p)^{2\times(54-i)} \qquad (1)$$

Formatted: Font: 14 pt

We also have 8 parity blocks but there is a particular pattern, 00 11 11 00 00 11 11 00, all these bits need to be correct then the probability of correct sync header pattern of parity blocks is,

$$\left((1-p)^2\right)^8 \qquad (2)$$

Formatted: Font: 14 pt

Formatted: Font: Times New Roman, 14 pt

Both of the above criteria need to be fulfill then the probability of true-lock is

Deleted: 2008-03-14

$$P_{lock} = \sum_{i=0}^{54} C_{54}^{i} (p^2)^i (1-p)^{2\times(54-i)} \times \left((1-p)^2\right)^8 \qquad (3)$$

Formatted: Font: 14 pt

$$Lock\_time = \frac{1}{P_{lock}} \times 62 \times 66 \times 10^{-10} (s) \qquad (4)$$

Formatted: Font: 14 pt

For p =10e-3, Lock_time = 0.47 us.

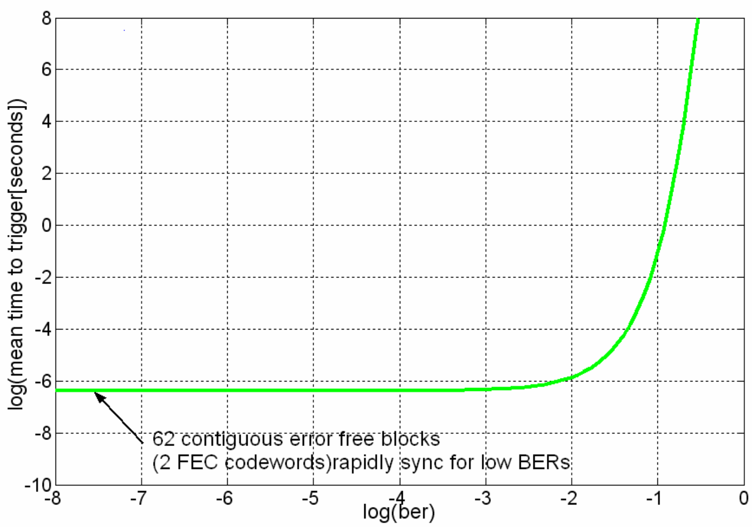

Figure 3. Performance of codeword synchronization

The curve is similar to the one in http://www.ieee802.org/3/10GEPON_study/public/july06/thaler_1_0706.pdf, which is for the Lock State machine in 802.3ae.

Deleted: 64/66b state machine

However, the above calculation is assuming the state machine is looking at the right position of the incoming codeword e.g 27 data blocks + 4 parity blocks, which is the best case for our Lock State Machine. If there is not any noise, ***e.g. p =0, Lock_time = 6****2×66 ×10*$^{-10}$*=0. 4092***us.**

A true MTT-true-Lock should be a statistics problem, we need to consider the average lock time of all possible incoming codeword positions, which the state machine is looking at.

Since there are too many cases, we don't have time to solve it right now. However we have run computer simulation, 300000 testing times for each BER point and calculate the average lock time of each BER point, for our state machine for true-lock time as follows,

Deleted: 2008-03-14

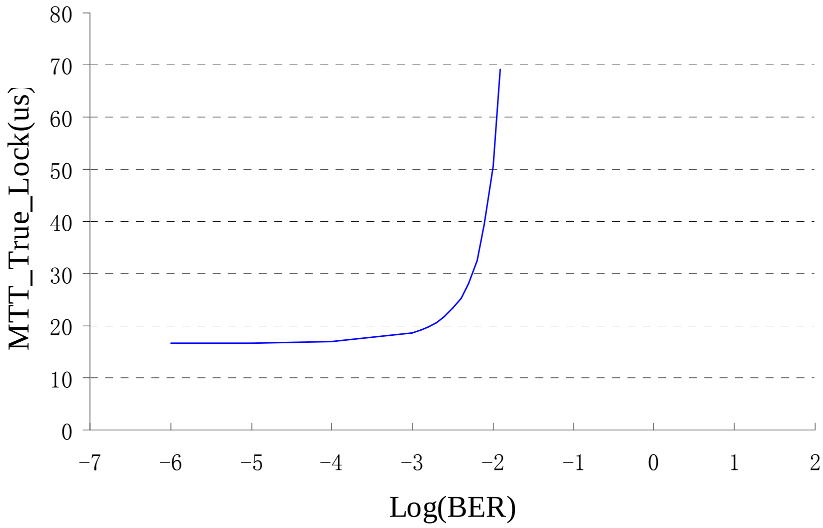

Figure 4. Computer simulation of MTT-true-lock of the Lock State Machine

For p=0, the MTT-true-lock ≈ 14 us.
For p=10e-3, the MTT-true-lock ≈ 18.6037 us.

Deleted: (300000 times of 1000 codewords per time has been tested)

Deleted: *6*

Deleted: (100000 times of 1000 codewords per time has been tested)

MTT-false-lock

There are three cases for false-lock to happen,

Case 1

The state machine finds the right block position e.g. (2 sync headers + 64 data, Figure 5. below), but wrong codeword position.

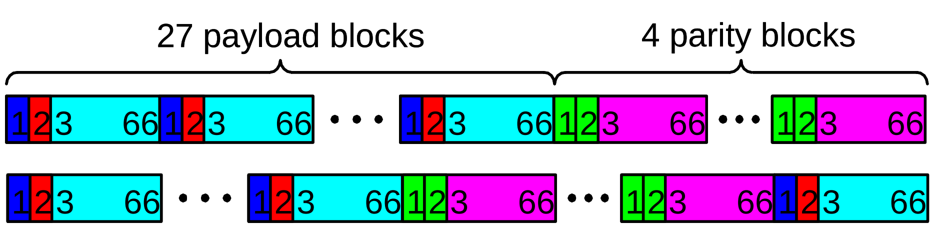

Figure 5. Schematic of false codeword alignment for Case 1

The incorrect sync headers have been changed to the correct sync headers due to BER. We have been proposed this to the group several times, 0 2 2 0 is better than the other sync headers pattern for the parity blocks.

Figure 5 shows the codeword (it should be two codewords) has been shifted **one**

Deleted: The above f

Deleted: 2008-03-14

block to left, and the false lock probability is (2 cases to left or to right)

$$\underbrace{p^{12}(1-p)^{8}}_{\text{Data blocks \& Parity blocks to cause false lock}} \quad \underbrace{\sum_{i=0}^{52} C_i^{52}[p^{2i}(1-p)^{2(52-i)}]}_{\text{52 correct Data blocks}} \qquad (5)$$

The false lock probability for shifting **two** blocks to left or to right is, (2 cases)

$$p^{16}(1-p)^{8}\sum_{i=0}^{50} C_i^{50}[p^{2i}(1-p)^{2(50-i)}] \qquad (6)$$

Three blocks to left or to right, (2 cases)

$$p^{12}(1-p)^{16}\sum_{i=0}^{48} C_i^{48}[p^{2i}(1-p)^{2(48-i)}] \qquad (7)$$

Four or more (up to 15) blocks to left or to right, (24 cases)

$$p^{16}(1-p)^{16}\sum_{i=0}^{52} C_i^{52}[p^{2i}(1-p)^{2(46-i)}] \qquad (8)$$

The total false lock probability due to sync headers errors is the sum of all above,

$$\begin{aligned} &2p^{12}(1-p)^{8}\sum_{i=0}^{52} C_i^{52}[p^{2i}(1-p)^{2(52-i)}] + 2p^{16}(1-p)^{8}\sum_{i=0}^{50} C_i^{50}[p^{2i}(1-p)^{2(50-i)}] \\ &+ 2p^{12}(1-p)^{16}\sum_{i=0}^{48} C_i^{48}[p^{2i}(1-p)^{2(48-i)}] + 24p^{16}(1-p)^{16}\sum_{i=0}^{52} C_i^{52}[p^{2i}(1-p)^{2(46-i)}] \end{aligned} \qquad (9)$$

Formatted: Font: 14 pt

Formatted: Font: 14 pt

Formatted: Font: 14 pt

Formatted: Font: 14 pt

Case 2

The state machine finds the wrong blocks position, the data blocks and the parity blocks has a identical pattern as the sync header patterns that the state machine is looking for, the figure below,

Deleted: 2008-03-14

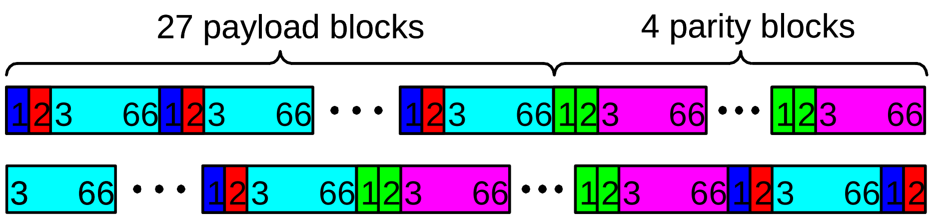

Figure 6. Schematic of false codeword alignment for Case 2

Since every bit of data payload or parity are considered random and i.i.d, therefore the probability of two data bits to be valid sync headers of data block ("01" or "10") is 0.5; the probability of two data bits to be valid sync headers of parity blocks (either "00" or "11") is 0.25.
Hence the false locking probability due to data payload or parity pattern, happen to be a valid sync headers pattern to the Lock State Machine is

$$63\times(\frac{1}{2})^{54}\times(\frac{1}{4})^{8}$$, which is corresponding to 243160 years (10)

Formatted: Font: 14 pt

There are 63 cases of case2.

Case 3

The state machine finds the wrong blocks position, one bit of the payload and one bit of sync headers create a false sync headers pattern, which is identical to the correct sync header patterns that the state machine is looking for as Figure 7.

Deleted: , the figure below

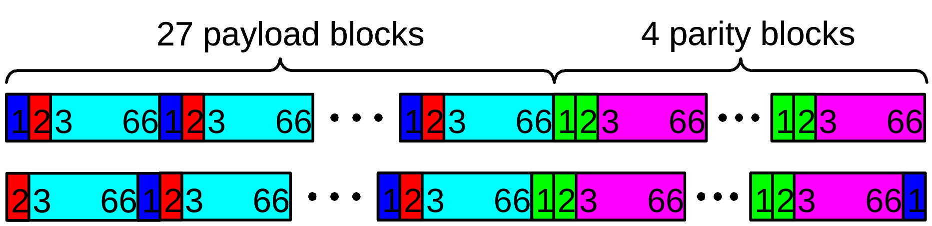

Figure 7. Schematic of false codeword alignment for Case 3

For the data blocks, the probability of shifted sync header bit to be the same as the right sync header bit is 0.5 (i.e. SH_1 = SH_2) as well as the first bit of the data payload. Since the two sync header bits of parity block are the same then we only need to consider the first bit of 64-bit parity to the same as the second sync header bit and the error probability is 0.5.
Hence the false locking probability due to shifted one bit is

Deleted: one parity

Deleted: 2008-03-14

$2\times(\frac{1}{2})^{54}\times(\frac{1}{2})^{8}$, which is corresponding to 29920 years (11)

There are 2 cases of case3.

The true MTT-false-lock probability is the total of all the above cases, which is as follows

$$P_{false\_lock}=2p^{12}(1-p)^{8}\sum_{i=0}^{52}[p^{2i}(1-p)^{2(52-i)}]+2p^{16}(1-p)^{8}\sum_{i=0}^{50}[p^{2i}(1-p)^{2(50-i)}]$$
$$+2p^{12}(1-p)^{16}\sum_{i=0}^{48}[p^{2i}(1-p)^{2(48-i)}]+24p^{16}(1-p)^{16}\sum_{i=0}^{52}[p^{2i}(1-p)^{2(46-i)}]\qquad(12)$$
$$+63\times(\frac{1}{2})^{70}+2\times(\frac{1}{2})^{62}$$

$$False\_lock\_time=\frac{1}{P_{false-lock}}\times 62\times 66\times 10^{-10}(s)\qquad(13)$$

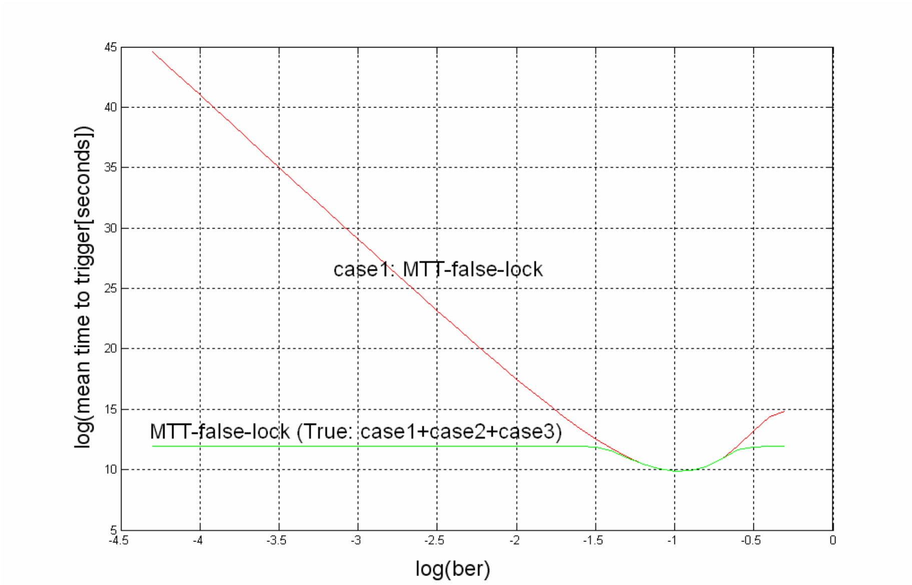

Figure 8. Performance of MTT-false-lock

The MTT-false-lock at p=10e-3 is 8.4e+011s, which is around 26636 years.

As in Figure 9, we are around 6 times better than 0 0 0 2. As you can see in Figure 9, the dominant term at BER@10e-3 is $2\times(\frac{1}{2})^{62}$. However, it also shows a gain by using 0 2 2 0 compares to 0 0 0 2.

Formatted: Font: 14 pt

Formatted: Font: 14 pt

Formatted: Font: 14 pt

Deleted: From the graph

Deleted: from the graph

Deleted: p =

Deleted: 2008-03-14

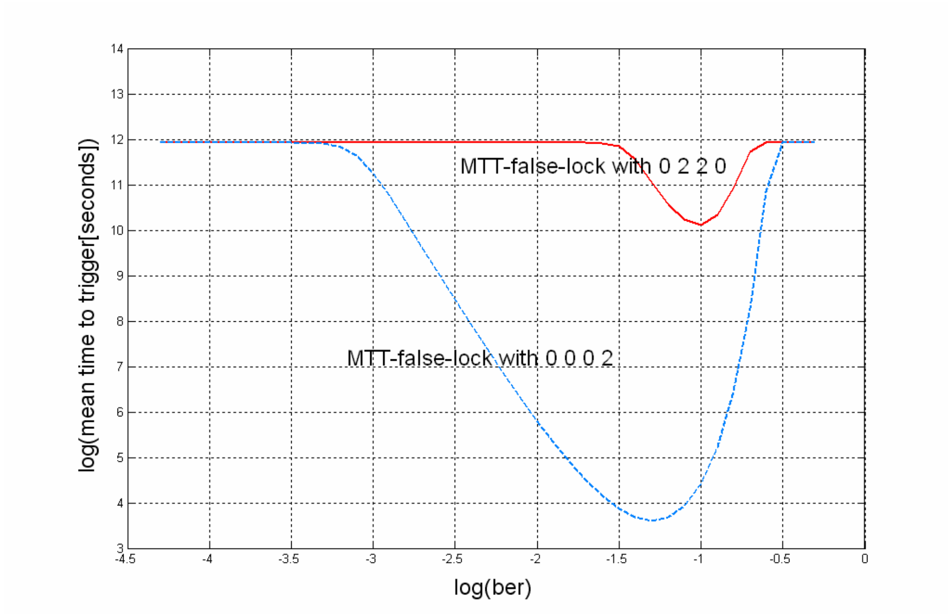

Figure 9. Performance comparison between "0 2 2 0" and "0 0 0 2"

MTT-unlock

There shall be two types of MTT-unlock: MTT-true-unlock and MTT-false-unlock.

MTT-true-unlock is from false-lock to unlock, we think that this is mainly caused by mis-alignment, we have showed three cases in the section of MTT-false-lock. Let's say that we are in a lock state and it is false locking, then the error probability of sync header bits of next incoming two codewords (62 continuous blocks) is 0.5 so the state machine will pick this up with a very high probability, in other word, this should happen very fast.

MTT-false-unlock is from true-lock to unlock, we think that this is mainly due to the BER of sync headers bits. In our case, we should consider BER@10e-3.

Deleted: caused

Deleted:

Deleted: p=

Our state machine only checks for the sync header bit errors up to a certain number (e.g. 16 or more in the current draft) to kick a lock state to an unlock state, i=16. It doesn't care if it is false-lock to unlock or true-lock to unlock.

And we should consider in two separate parts, the number of error sync headers, j, in the 54 data blocks and the number of error sync headers, i-j, in 8 parity blocks.

The probability of j number of error sync headers in 54 data blocks is (only one bit in error to cause false sync headers for data blocks),

Deleted: 2008-03-14

$$C_{54}^{j}(2p(1-p))^{j}(1-2p(1-p))^{54-j} \qquad (14)$$

Formatted: Font: 14 pt

Formatted: Left

The probability of i-j number of error sync headers in 8 parity blocks is (one bit or two bits in error to cause false sync headers for parity blocks),

$$C_{8}^{i-j}(1-(1-p)^{2})^{i-j}(1-p)^{2\times(8-i+j)} \qquad (15)$$

Now considering all the combination and the two parts together, we have the following probability for unlock.

$$P_{\_unlock} = \sum_{i=x}^{62}\sum_{j=i-8}^{t_0}[C_{54}^{j}(2p(1-p))^{j}(1-2p(1-p))^{54-j} \times C_{8}^{i-j}(1-(1-p)^{2})^{i-j}(1-p)^{2\times(8-i+j)}], t_0 = \begin{cases} i, & i<54 \\ 54, & i>=54 \end{cases} \qquad (16)$$

Formatted: Font: 14 pt

$$Unlock\_time = \frac{1}{P_{unlock}} \times 62 \times 66 \times 10^{-10}(s) \qquad (17)$$

Formatted: Font: 14 pt

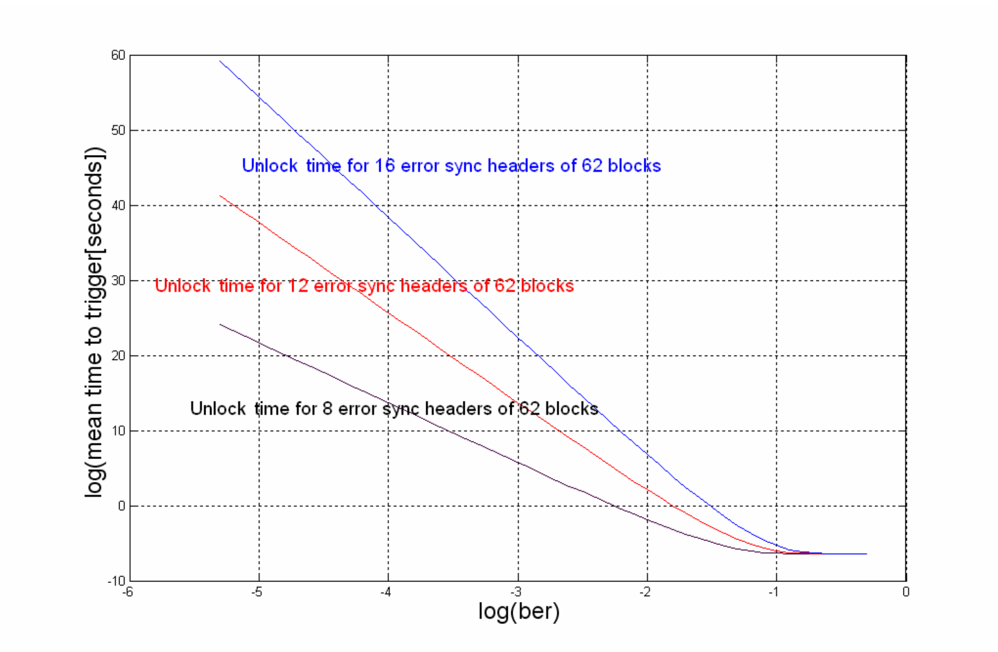

Figure 10. Performance of MTT-unlock

MTT-true-unlock

The BER can be considered as 0.5 as the sync header of the next incoming two

Deleted: 2008-03-14

codewords can be considered as random and i.i.d,

If we drop out of lock after only 8 errors then
MTT-true-unlock = 0.4092 us. (4. 092000000008596e−007s)

If we drop out of lock after only 12 errors then
MTT-true-unlock = 0.4092 us. (4. 092000020952663e−007s)

If we drop out of lock after only 16 errors then
MTT-true-unlock = 0.4092 us. (4. 092008748388514e−007s)

MTT-false-unlock
The BER is @10e-3,

Deleted: p =

If we drop out of lock after only 8 errors then
MTT-false-unlock = 5.243e5s. (5. 242701713634630e+005s)

If we drop out of lock after only 12 errors then
MTT-false-unlock = 5.13e13s ≈1.63 million years.
(5.129158133079716e+013)

If we drop out of lock after only 16 errors then
MTT-false-unlock = 2.528e22 s, which is longer than life of universe.
(2.528433351458057e+022)

Additional FEC Decoder Locking Kickout Feature
This section considers the performance of the extra FEC decoder kickout feature, which addresses the pathological case of a false lock which can happen if the receiver slips exactly one block after successfully locking.

We have considered how to define the number of FEC decode failures to kick a lock state to an unlock state.
The way to calculate the MTT_unlock is

$$MTT\_unlock = \frac{1}{(P)^m} \times m \times 62 \times 66 \times 10^{-10} \qquad (18)$$

Formatted: Font: Times New Roman, 14 pt

P is the codeword error probability of an FEC.

There are two types of unlock, True_unlock and False_unlock.

Deleted: ¶

False_unlock
The state machine has locked correctly and we don't want the FEC to kick a **truely lock state** to an **unlock state** due to FEC decode failure. This is mainly

Deleted: 2008-03-14

due to the FER (Frame Error Rate) or the codeword error probability of an FEC, which is a function of the input BER.

For the RS(255,223), an input BER@10e-3 is corresponding to an FER@10e-11 after FEC decode.

$$MTT\_false\_unlock = \frac{1}{(FER)^m} \times m \times 62 \times 66 \times 10^{-10} \geq 3.17 \times 10^{17} (\text{life of universe}) \quad (19)$$

Formatted: Font: Times New Roman, 14 pt

m >= 3

Deleted: 1

Deleted: ¶

True unlock

The state machine has locked incorrectly and we want the FEC to kick a **falsely lock state** to an **unlock state** very fast. This is mainly due to randomly generated X will be interpreted as a codeword C after RS(255,223) decoding, which should not happen.

Assuming *P* is the probability with that a randomly generated X will be interpreted as a codeword C after RS(255,223) decoding, then

Formatted: Font: Italic

Deleted: which is

$$P = \frac{2^{km}}{2^{nm}} \cdot \sum_{i=0}^{t} \binom{n}{i} (q-1)^i = \frac{2^{km}}{2^{nm}} \cdot \sum_{i=0}^{t} \binom{n}{i} (q-1)^i \frac{2^{223\times 8}}{2^{255\times 8}} \sum_{i=0}^{16} \binom{255}{i} (256-1)^i = 2.6 \times 10^{-14} \quad (20)$$

Deleted: ¶

Formatted: Font: Times New Roman, 14 pt

Hence, the probability that X will not be decoded as a correct RS code C is 1-*P*.

Considering that we want the false-unlock to happen very slow (life time of universe) and the true-unlock to happen very fast. Thus, we should check the MTT-true-unlock when m=3,

$$MTT\_true\_unlock = \frac{1}{(1-P)^3} \times 3 \times 62 \times 66 \times 10^{-10} = 1.23 \times 10^{-6} \text{ sec} \quad (21)$$

Formatted: Font: Times New Roman, 14 pt

Conclusion

Considering the above two MTT_unlock and our problem, "In this failure scenario, our state machine will only see 4 'wrong' sync headers per codeword, or 8 errors per '62 block' section. Since our machine expects to use 16 errors as the criterion for dropping out of lock, we will be stuck in this bogus alignment forever. "

If the above mentioned problem happen, then we will need to see **three** consecutive FEC decode failure to kick a lock state to an unlock state.

Deleted: 2008-03-14