RCWMS in RSFEC EEE

Don Cober – CoMIRA Solutions Inc.

Oded Wertheim – Mellanox

Supporters

Background

- Draft 1.0 defines a fast lock mechanism based on known pattern detection by sending unscrambled data over the channel during the WAKE periods.
- Several comments on D1.0 (#38, #69, #70, #74, #75, #169, #178, #179, #180) identify issues related to the fast lock mechanism based on scrambler_bypass.
- We propose an alternative solution for rapid codeword lock using Rapid Codeword Markers (RCWMs)
- The RCWMs solution provides a simple remedy for the issues that were identified.

RS-FEC EEE Rapid Lock Issues

- Unscrambled IDLEs/LPIs are not DC balanced and don’t have enough transition density, which is unsuitable for the electrical signaling.
  - See ran_042915_25GE_adhoc and #38,#179
- Sending unscrambled IDLEs/LPIs enables the peer port to detect transcoding block boundaries. However it does not provide an efficient way to detect the FEC codeword boundaries.
  - Comment #179
- Sending unscrambled data at the start of TX_WAKE (TX_MODE transition from ALERT to DATA), does not allow the peer PMA/PMD to refresh. As a result the peer RS-FEC may not detect the pattern and lock.
  - Comment #75

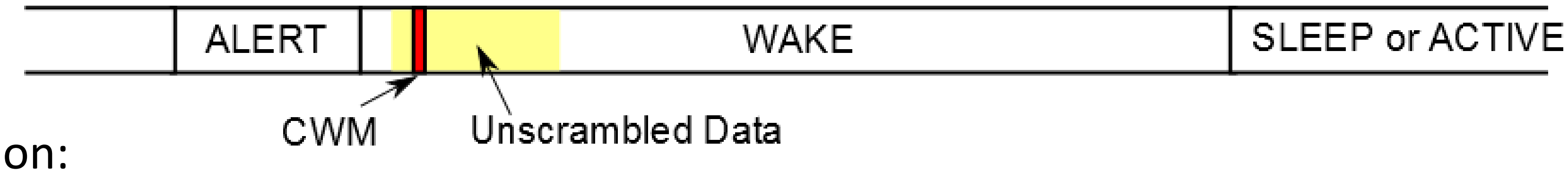
RS-FEC EEE Rapid Lock Issues

- First CWM transmission (comment #180)
  - The transmitter sends unscrambled data for 0.9µs - 1.1µs.
  - The transmitter inserts the first codeword marker at the beginning of the second codeword - 0.2 µs - 0.4 µs after the scrambler bypass.
  - The receiver may not succeed to identify the codeword boundaries in time and miss the codeword marker.

- The rapid codeword lock does not describe how to lock on the codeword marker position.
  - Comment #178

Suggested Remedy - TX

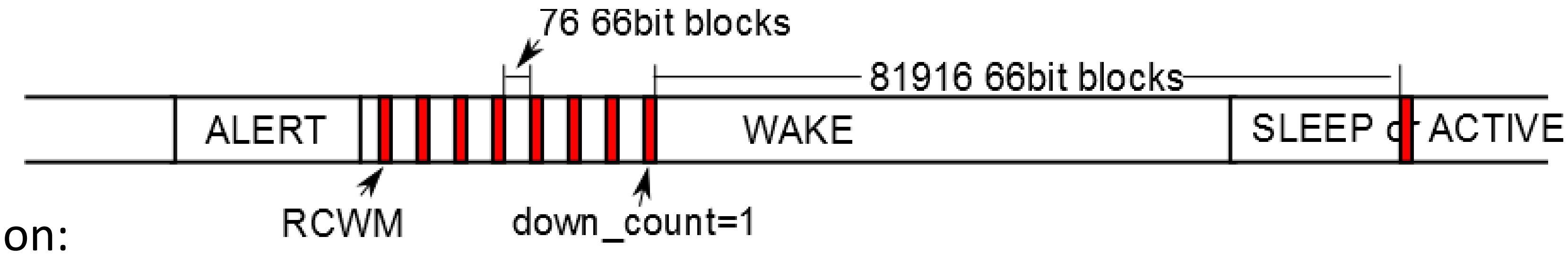
Current Specification:



On TX at ALERT->DATA transition:

1. Send unscrambled data for a period of 1.0-1.1 µs
2. Start sending CWM at the second full codeword.

Proposed:



On TX at ALERT->DATA transition:

1. Insert 40 RCWMs with down_count of 40…1 on the first 40 codewords
2. Transition back to normal CWMs with a gap of 1023 CWs between the last RCWM and the first CWM

Suggested Remedy - RX

Current Specification:

On RX at QUIET->DATA transition:

1. Search for codeword boundary using unscrambled data.
2. Detect transition from unscrambled to scrambled data to infer transition from RCWMs to CWMs

Proposed:

On RX at QUIET->DATA transition:

1. Search for codeword boundary using RCWMs. (Lock after receiving two good RCWMs)
2. Use down_count value to predict transition from RCWMs to CWMs

RCWM Solution Benefits

- Removes the need to send unscrambled data. (data sent in EEE mode is the same as data normally sent in DC balance and transition density)
- Does not require changes to the PCS or additional signaling for the FEC.
- Re-uses well understood ideas from 802.3bj for rapid lock
- Allows a wider window of time for the FEC to lock.

RCWM Structure

0 7	8 15	16 23	24 31	32 39	40 47	48 55	56 63	64 71	72 79	80 87	88 95	96 103	104 111	112 119	120 127
0xC1	0x68	0x21	CD3	0x3E	0x97	0xDE	CD7	0x9D	0x71	0x8E	CD3	0x62	0x8E	0x71	CD7

128 135	136 143	144 151	152 159	160 167	168 175	176 183	184 191	192 199	200 207	208 215	216 223	224 231	232 239	240 247	248 255	256
0x59	0x4B	0xE8	CD3	0xA6	0xB4	0x17	CD7	0x4D	0x95	0x7B	CD3	0xB2	0x6A	0x84	CD7	0

- RCWM is the same as CWM except:
  - bits [24:31] are set to down_count instead of 0x33
  - bits [56:63] are set to the bitwise inversion of down_count instead of 0xCC
  - bits [88:95] are set to down_count instead of 0x33
  - bits [120:127] are set to the bitwise inversion of down_count instead of 0xCC
  - bits [152:159] are set to down_count instead of 0x33
  - bits [184:191] are set to the bitwise inversion of down_count instead of 0xCC
  - bits [216:223] are set to down_count instead of 0x33
  - bits [248:255] are set to the bitwise inversion of down_count instead of 0xCC

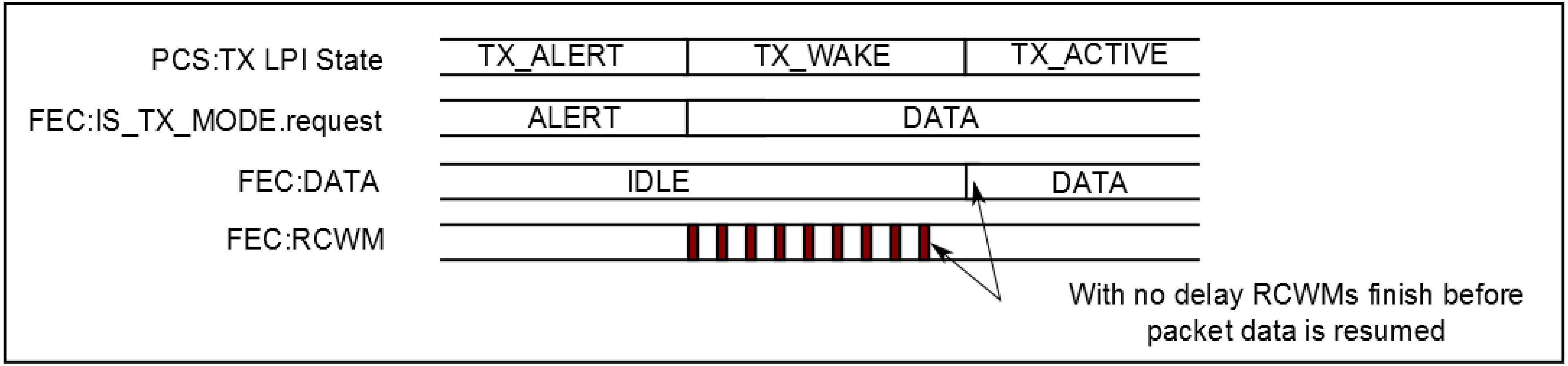
RCWM frequency.

- We propose to send RCWMs on every codeword during the TX_WAKE state. The justification for this is:
  1. With more RCWMs sent, locking is easier.
     - A locking mechanism could potentially test fewer positions per codeword and still have the same effect as a more complex search over more sparse RCWMs.
  2. With more RCWMs sent, locking is more reliable.
     - Since some of the RCWMs will be lost to errors and link up delay, having more RCWMs to search for will allow the link to wake up under harsher conditions.
  3. Since no data is being transferred, nothing is lost by removing more idles
  4. The RCWMs are roughly DC balanced (129 zeros, 128 ones) and have similar transition density to the scrambled idles they replace.

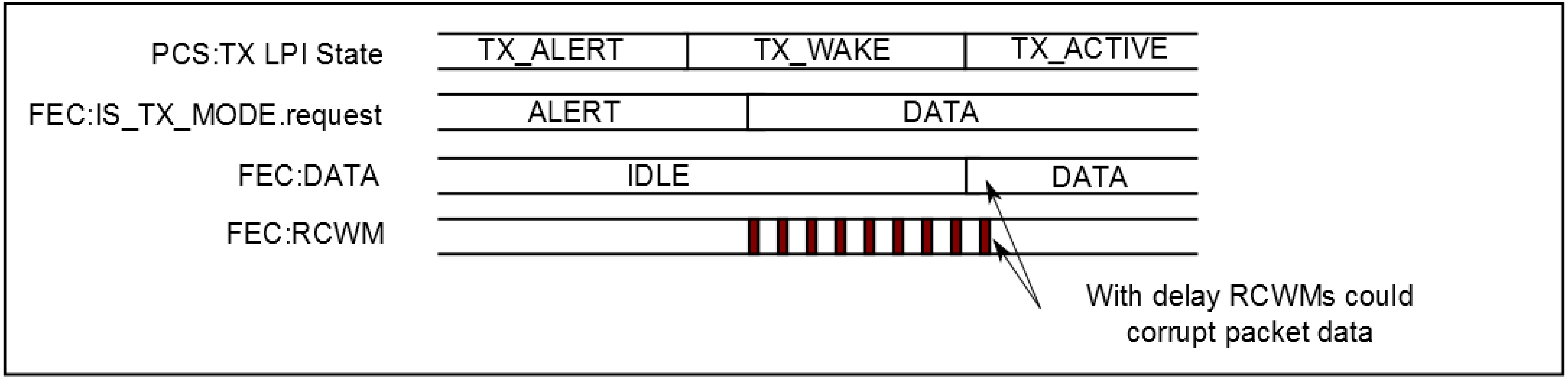
RCWM count

- We propose to send 40 RCWMs after the transition from QUIET to DATA. This is justified because:
  1. As stated in the previous slide, it is desirable to send the largest number of RCWMs possible.
  2. We must ensure that RCWMs are not sent after the PCS's TX_WAKE state
     - If RCWMs are sent during the TX_ACTIVE state, then the idle insertion/deletion would be frequent enough to corrupt packets
  3. Each RCWM requires 0.2048 µs and the length of the PCS TX_wake is 10.9 µs, but we must also take into account the delay of the LPI primitive when using a FEC connected through the C2C defined in annex 109A. The delay may cause the FEC to detect the transition from QUIET to DATA some time after the PCS has entered the WAKE state.
     - 40 RCWMs = 8.192 µs which allows up to 2.7 µs to account for request primitive delay.

Example of request primitive delay



With no delay RCWMs finish before packet data is resumed



With delay RCWMs could corrupt packet data

CWM Lock for RCWMs

- CWM Lock is gained when two valid consecutive RCWMs are seen.
    - A valid RCWM has 9/12 correct nibbles in fields [0:23] and [32:55] (This is the same as the cwm_valid condition for normal CWMs)
    - Valid consecutive RCWMs have decrementing down counts.

- FEC_align_status is set to true after the second RCWM

- The valid down_count value in the second RCWM can be used to predict the transition to normal CWMs. (A normal codeword will be seen down_count+1023 CWs from now)

Suggested Remedy - Summary

- Send Rapid Codeword Markers (RCWMs) to enable the peer port to rapidly achieve codeword marker lock.

- Use RS-FEC EEE fast lock as defined in D1.0, with the following changes:
  - TX: Instead of bypassing the scrambler, set down_count = 40 and insert a RCWM in the beginning of each of the following 40 FEC codewords.
  - RX: Instead of performing rapid lock based on a known pattern, lock on the codeword marker position based on RCWMs.

- The conditions for sending / searching for RCWMs are the same as defined in D1.0 for the rapid lock:
  - Tx: TX_MODE Send RCWMs on the transition from ALERT => DATA
  - Rx: RX_MODE (RX_TX_MODE) search for RCWMs on the transition from QUIET=> DATA

- RCWM are identical to regular CWMs with the down_count value set in offsets: 24:31 and the bit-wise inversion of the down_count in offsets 56:63.

Suggested Remedy – Proposed Changes to the Draft

1. Modify 108.5.2.7 RS-FEC encoding for rapid codeword lock (EEE deep sleep) (page 105)
   - **Replace a), b) (lines 5-9) with:**

     a) Set down_count to 40 and send 40 rapid codeword markers (RCWMs). This causes the Codeword marker insertion function (108.5.2.4) to insert a RCWM in the beginning of each of the following 40 FEC codewords.

     b) The first regular codeword marker is inserted at the beginning of the 1024th RS-FEC codeword after the RCWM with down_count = 1.

   - **Remove lines 14-17 "As a result … by the remote PCS“**

2. Add Rapid Codeword marker insertion description:
   - **Add at the bottom of 108.5.2.4 Codeword marker insertion: (page 105, line 37)**

     For the optional EEE capability, a rapid method of FEC alignment is used when operating in the deep sleep low power state using Rapid Codeword Markers (RCWMs). RCWMs are inserted at the beginning of 40 codewords following the transmitter tx_mode transition from ALERT to DATA.

     RCWMs are identical to regular CWMs with the exception that the constant value of 0x33 in offsets 24:31, 88:95, 152:159, 216:223 is replaced with a down_count value, and the constant value of 0xCC in offsets 56:63, 120:127, 184:191, 248:255 is replaced with the bit-wise inversion of the down_count. The down_count is decremented each time a RCWM is sent.

Suggested Remedy – Proposed Changes to the Draft

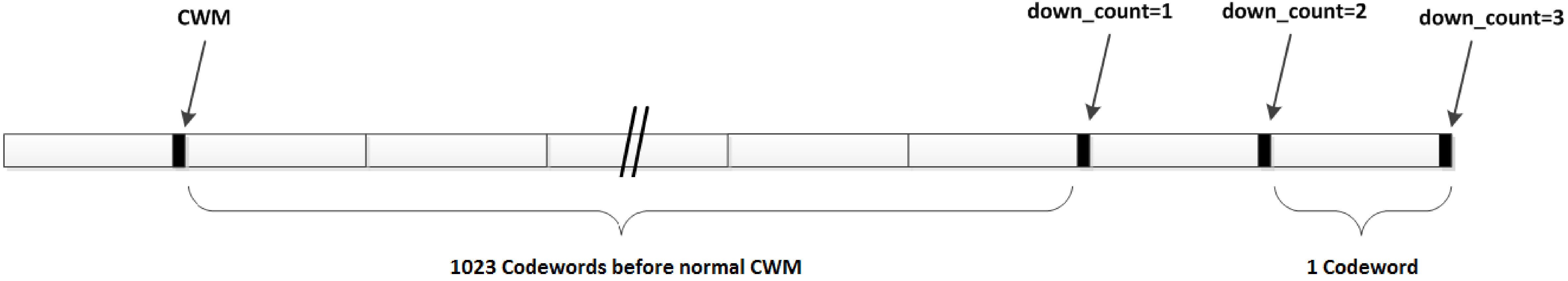


Figure 108-3 – RCWM transition

0 7	8 15	16 23	24 31	32 39	40 47	48 55	56 63	
M_0	M_1	M_2	CD_3	M_4	M_5	M_6	CD_7	
64 71	72 79	80 87	88 95	96 103	104 111	112 119	120 127	
M_0	M_1	M_2	CD_3	M_4	M_5	M_6	CD_7	
128 135	136 143	144 151	152 159	160 167	168 175	176 183	184 191	
M_0	M_1	M_2	CD_3	M_4	M_5	M_6	CD_7	
192 199	200 207	208 215	216 223	224 231	232 239	240 247	248 255	256
M_0	M_1	M_2	CD_3	M_4	M_5	M_6	CD_7	0

Figure 108-4 – RCWM format

Suggested Remedy – Proposed Changes to the Draft

3. In 108.5.3.6 Rate compensation for codeword markers in the receive direction, (page 109, line 9)

   Replace: "Insert idle characters, according to the rules in 49.2.4.7, to fill in as necessary for any deleted codeword markers."
   With: "Insert idle or low power idle (LPI) characters, according to the rules in 49.2.4.7, to fill in as necessary for any deleted codeword markers or rapid codeword markers.“

4. In 108.5.3.7 Rapid codeword lock for EEE deep sleep: (page 109)
   - **Remove a) in line 29.**
   - **Replace c) in line 33 with:**

     c) Enable the RS-FEC rapid codeword lock mechanism, which attempts to detect rapid codeword markers sent by the remote RS-FEC transmit function (see 108.5.2.7). When two sequential rapid codeword markers are detected, the start location of the RS-FEC codeword is set to the start location of the rapid codeword markers. The next codeword marker position is set to 1024 codewords following the rapid codeword marker with down_count = 1.

   - **Remove the sentence in line 44: "When the decoding .. Set to false“**
   - **Replace 1) in line 50 "Two 64B/66B .. true to false" with:**

     1) The RS-FEC codeword monitor state diagram (Figure 108-6) reaches the CW_GOOD state.

Suggested Remedy – Proposed Changes to the Draft

5. In 108.5.4.2 State variables:
   - **Remove page 110, lines 35-40: descrambler_bypass**
   - **Remove page 110, lines 52-54: scrambler_bypass**
   - **Remove page 112, lines 1-3: "optional EEE … always false."**

   - **Add a new variable to page 100, lines 35-40:**

     down_count

     A counter that is used in rapid codeword markers and is decremented each time a RCWM is sent. The counter initial value is set by the RS-FEC transmit function when the tx_mode parameter of the FEC:IS_TX_MODE.request primitive from ALERT to DATA.