Clause 161 AM Lock Issue

Jeff Slavick

803.2ck ad-hoc Jan 8, 2020

BROADCOM®

Contributors

- Shawn Nicholl
- Ben Jones

Copper Interface Bring up Options

- Auto-negotiation
  - Devices determine highest BW link both sides advertise and operate at it to set the SerDes rate and PCS/FEC mode
  - Link Training is run to establish TxEq and precoding
- Forced Rate (AUI style bringup)
  - User sets configuration (Serdes rate, PCS/FEC mode, TxEq, precoding)
    - Link Training should be run for PAM4 links, but usually is skipped for NRZ links
  - This mode is a supported mode by the standard because AN is optional to be used

73.1 Auto-Negotiation introduction

While implementation of Auto-Negotiation is mandatory for Backplane Ethernet PHYs, the use of Auto-Negotiation is optional. Parallel detection shall be provided for legacy devices that do not support Auto-Negotiation.

Debugging Link up Issues

- If PCS/FEC mode is not set the same on both sides then the link will fail to come up.
- To debug LinkUp issues you will start at the failing layer and work your way down to determine what is failing.
- For example to figure out why LinkUp is not occurring you would check following in this order.
  1) PCS::PCS_status
  2) PCS::alignment_valid
  3) PCS::am_lock[19:0]
  4) FEC::fec_alignment_valid
  5) FEC::amps_lock[3:0]
  6) SERDES::training_failure

Issue with CI161 AM pattern

- If one side is in CI91 mode and the other in CI161 mode then the status looks like

Status	Co-located	Separated
PCS_status	?	FALSE
PCS_align_valid	?	FALSE
PCS_am_lock	?	FALSE
FEC_alignment_valid	TRUE/TOGGLE	TRUE/TOGGLE
FEC_amps_lock	TRUE	TRUE
SERDES_training_failure	FALSE	FALSE

- FEC frames but is getting consecutive uncorrectable codewords and re-framing constantly which prevents the PCS from coming up.

Debugging Issue

- Uncorrectable codewords is usually caused by poor SerDes performance
  - However SerDes eye metrics would appear to be “good”
- Checking loopback configuration successfully comes up
  - both sides of local port have same FEC mode
- Changing cables does not resolve the issue
  - still have cross wired modes

Historical operations

- When multiple FEC modes are supported for a given baud rate and encoding they aren't allowed to achieve alignment to each other when the modes are not the same.
  - 26.5625Gbaud PAM4 is used for:
    - 400Gx8
    - 200Gx4
    - 100Gx2
    - 50Gx1
  - 25.78125GBaud NRZ is used for:
    - 100Gx4
    - 25Gx1 (3 different FEC modes)
- None of the above will achieve FEC frame to one another
- This is not true for the current state of CI161

RS-FEC Alignment Markers

FEC mode	Clause	Lanes	CM	1st UM Lane0	Interval	Bit Interval
400G – RS544 Int	119	16	0x9A 4A 26 *CI119 CM*	0x01 71 F3 *400G – L0*	163,840 257b 8192 CW	2,785,280
200G – RS544 Int	119	8	0x9A 4A 26 *CI119 CM*	0xB3 C0 8C *200G – L0*	82,920 257b 4096 CW	2,785,280
100G – RS544 Int	161	4	0xC1 68 21 *100G – L0*	0xF5 07 09 *100G – L4*	81,920 257b 4096 CW	5,570,560
100G – RS544	91	4	0xC1 68 21 *100G – L0*	0xF5 07 09 *100G – L4*	81,920 257b 4096 CW	5,570,560
100G - EEE	91	4	0x3E 97 DE *~100G – L0*	0x0A F8 F6 *~100G – L4*	40 257b 2 CW	2640
100G – RS528	91	4	0xC1 68 21 *100G – L0*	0xF5 07 09 *100G – L4*	81,920 257b 4096 CW	5,406,720
50G – RS544	134	2	0x90 76 47 *40G – L0*	0xC5 65 9B *40G – L2*	20,480 257b 1024 CW	2,785,280
25G – RS528	108	1	0xC1 68 21 *100G – L0*	0xF0 C4 47 *40G – L1*	20,480 257b 1024 CW	5,406,720

Possible paths forward

A. Prevent Legacy and CI161 from achieving AM lock to each other
   - Link doesn't come up
   - Most robust for system, both sides can't comprehend each other

B. Prevent CI161 from achieving AM lock to Legacy
   - Legacy side would flap continuously, CI161 would never PCS/FEC Lock
   - Prevents flapping on one side

C. Provide indicator CI161 is getting wrong format
   - Flaps occur and user has to know what to look for

D. Do Nothing

Recommendation

- Implement Option A
  - It's the most robust solution
  - Users get a response from the link they're use to when mis-configuration occurs
- Next few slides shows several methods to implement Option A
- Backup slides have methods for implementing Option B and C

Option A: Methods to Prevent False lock on both sides

- Change Common Marker (common pattern sent on all lanes)
  - Prevents both sides from locking
  - No longer same pattern used for AM lock for 100G PHYs
- Change Unique Marker (pattern used to identify which lane is which)
  - Prevents both sides from being able to re-order
  - New pattern to look for on every lane
- Change Both
  - Prevents both sides from locking
  - Requires analysis of new pattern to be done for Baseline wander

Option A: Methods to Prevent False Lock Implementation Choices

1) Change Common Marker
   - Use 40G Lane0 like 50G does
2) Change Common Marker
   - Use 200/400G Common Marker like the other RS-FEC Interleave
3) Change Unique Marker
   - Use 100G EEE format (Flip flop M0,1,2 and M4,5,6 for just UM patterns, AM4-19)
4) Change Both Common and Unique Markers
   - Full new AM patterns

Option A: Methods to Prevent False Lock Choices

	FEC mode	Lanes	CM	1st UM Lane0	Interval	Bit Interval
1	100G – RS544 Int	4	0x90 76 47 *40G – L0*	0xF5 07 09 *100G – L4*	81,920 257b 4096 CW	5,570,560
2	100G – RS544 Int	4	0x9A 4A 26 *Cl119 CM*	0xF5 07 09 *100G – L4*	81,920 257b 4096 CW	5,570,560
3	100G – RS544 Int	4	0xC1 68 21 *100G – L0*	0x0A F8 F6 *~100G – L4*	81,920 257b 4096 CW	5,570,560
4	100G – RS544 Int	4	0xAE C9 44	0x3C F2 48	81,920 257b 4096 CW	5,570,560
	100G – RS544	4	0xC1 68 21 *100G – L0*	0xF5 07 09 *100G – L4*	81,920 257b 4096 CW	5,570,560
	100G - EEE	4	0x3E 97 DE *~100G – L0*	0x0A F8 F6 *~100G – L4*	40 257b 2 CW	2640
	100G – RS528	4	0xC1 68 21 *100G – L0*	0xF5 07 09 *100G – L4*	81,920 257b 4096 CW	5,406,720
	50G – RS544	2	0x90 76 47 *40G – L0*	0xC5 65 9B *40G – L2*	20,480 257b 1024 CW	2,785,280

Option A Recommendation

- Do option A.2 or A.3
  - Change the Common Marker to 200/400G pattern (A.2)
  - Change the Unique Marker to 100G EEE pattern for AM4-19 (A.3)

A.2 text (change common marker to Cl119 format)

For *x*=0 to 19, amp_tx_*x*<63:0> is constructed as follows:

a) if x <= 3 amp_tx_*x*<23:0> is set to CM0, CM1, and CM2 as shown in Figure 119-4 (bits 23 to 0) using the values in Table 119-1 for PCS lane number x. if x >= 4 amp_tx_x<23:0> is set to M0, M1, and M2 as shown in Figure 82–9 (bits 25 to 2) using the values in Table 82–2 for PCS lane number x.

b) amp_tx_*x*<31:24> = am_tx_*x*<33:26>

c) if x <= 3 amp_tx_*x*<55:32> is set to CM4, CM5, and CM6 as shown in Figure 119-4 (bits 55 to 32) using the values in Table 119-1 for PCS lane number x. if x >= 4 amp_tx_x<55:32> is set to M4, M5, and M6 as shown in Figure 82–9 (bits 57 to 34) using the values in Table 82–2 for PCS lane number x.

d) amp_tx_*x*<63:56> = am_tx_*x*<65:58>

A.3 text (change AM4-19 to inverted, EEE format)

For *x*=0 to 19, amp_tx_*x*<63:0> is constructed as follows:

a) if x <= 3 amp_tx_*x*<23:0> is set to M0, M1, and M2 as shown in Figure 82–9 (bits 25 to 2) using the values in Table 82–2 for PCS lane number 0. if x >= 4 amp_tx_x<23:0> is set to M4, M5, and M6 as shown in Figure 82–9 (bits 57 to 34) using the values in Table 82–2 for PCS lane number x.

b) amp_tx_*x*<31:24> = am_tx_*x*<33:26>

c) if x <= 3 amp_tx_*x*<55:32> is set to M4, M5, and M6 as shown in Figure 82–9 (bits 57 to 34) using the values in Table 82–2 for PCS lane number 0. if x >= 4 amp_tx_x<55:32> is set to M0, M1, and M2 as shown in Figure 82–9 (bits 57 to 34) using the values in Table 82–2 for PCS lane number x.

d) amp_tx_*x*<63:56> = am_tx_*x*<65:58>

Backup

BROADCOM®

Issue with current Cl161 AM scheme

- Two rows contain the same data

FEC mode	Clause	Lanes	CM	1st UM Lane0	Interval	Bit Interval
100G – RS544 Int	161	4	0xC1 68 21 *100G – L0*	0xF5 07 09 *100G – L4*	81,920 257b 4096 CW	5,570,560
100G – RS544	91	4	0xC1 68 21 *100G – L0*	0xF5 07 09 *100G – L4*	81,920 257b 4096 CW	5,570,560

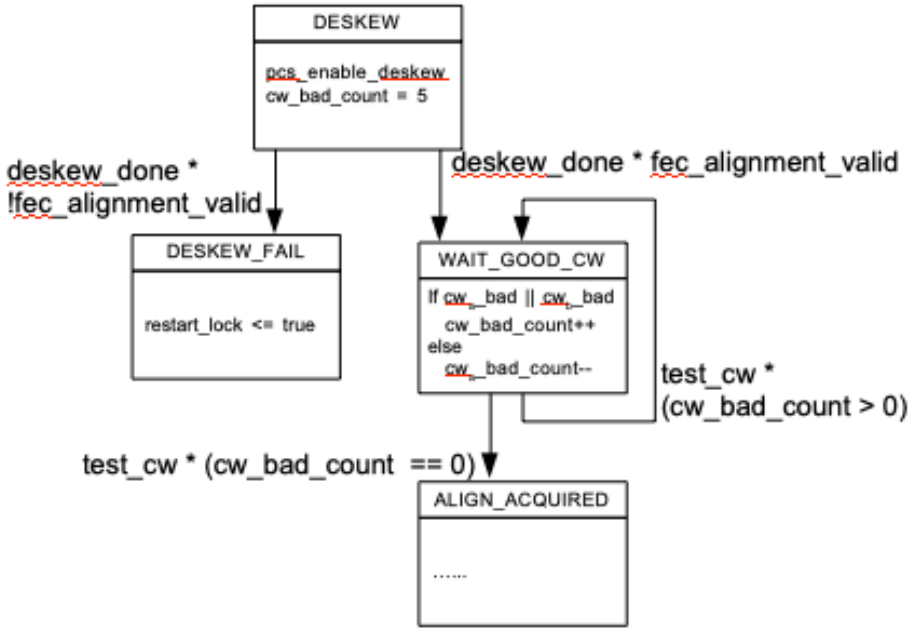
- Link Flap can continuously occur if one end sends interleave mode and other is non-interleave mode.
- FEC alignment lock scheme will frame, but when they decode the data the information is uncorrectable causing link to immediately go back down.

New AM patterns courtesy of Shawn and Ben

PCS lane number	Encoding[a] $\{M_0,M_1,M_2,BIP_3,M_4,M_5,M_6,BIP_7\}$	PCS lane number	Encoding[a] $\{M_0,M_1,M_2,BIP_3,M_4,M_5,M_6,BIP_7\}$
0	0xAE, 0xC9, 0x44, BIP3, 0x51, 0x36, 0xBB, BIP7	10	0x04, 0xAF, 0x37, BIP3, 0xFB, 0x50, 0xC8, BIP7
1	0xAE, 0xC9, 0x44, BIP3, 0x51, 0x36, 0xBB, BIP7	11	0x0C, 0xF3, 0xB4, BIP3, 0xF3, 0x0C, 0x4B, BIP7
2	0xAE, 0xC9, 0x44, BIP3, 0x51, 0x36, 0xBB, BIP7	12	0xFF, 0x58, 0x61, BIP3, 0x00, 0xA7, 0x9E, BIP7
3	0xAE, 0xC9, 0x44, BIP3, 0x51, 0x36, 0xBB, BIP7	13	0x48, 0xAC, 0x8A, BIP3, 0xB7, 0x53, 0x75, BIP7
4	0x3C, 0xF2, 0x48, BIP3, 0xC3, 0x0D, 0xB7, BIP7	14	0x17, 0xA1, 0xF7, BIP3, 0xE8, 0x5E, 0x08, BIP7
5	0xAF, 0x13, 0x52, BIP3, 0x50, 0xEC, 0xAD, BIP7	15	0x57, 0xA1, 0xE2, BIP3, 0xA8, 0x5E, 0x1D, BIP7
6	0x64, 0xF4, 0x6C, BIP3, 0x9B, 0x0B, 0x93, BIP7	16	0x37, 0x75, 0xCB, BIP3, 0xC8, 0x8A, 0x34, BIP7
7	0x64, 0x7A, 0x2F, BIP3, 0x9B, 0x85, 0xD0, BIP7	17	0x17, 0x11, 0x29, BIP3, 0xE8, 0xEE, 0xD6, BIP7
8	0xDB, 0x81, 0xEC, BIP3, 0x24, 0x7E, 0x13, BIP7	18	0x42, 0xA6, 0x3B, BIP3, 0xBD, 0x59, 0xC4, BIP7
9	0x3A, 0x50, 0xDC, BIP3, 0xC5, 0xAF, 0x23, BIP7	19	0x18, 0x85, 0x37, BIP3, 0xE7, 0x7A, 0xC8, BIP7

Option B: Prevent CI 161 from false lock

- Modify Alignment Lock State machine to check for correctable codewords before declaring lock to downstream logic



BROADCOM

Option C: Provide indicator of false lock

- Change the 5 Pad bits to differentiate between the two modes
  - In CI91 the 5b pad is 00101 pattern that is inverted every other frame.
  - Could change the pattern or change the inversion rate
  - Add logic that monitors the pad bits to check for it's pattern
  - Add MDIO register indicating which format is being received
- Would exist in CI161 capable devices, existing products for CI91 wouldn't have this
  - Could add an optional monitor to CI91 if we choose this path, so future products operating in CI91 mode could check for it

BROADCOM®

connecting everything®