Implementing a QSFP Cable Common Mistakes and Lessons Learned

Brent Rothermel, QLogic

- Introduction
- Mechanical Pitfalls
- I2C Pitfalls
- Memory Map Pitfalls

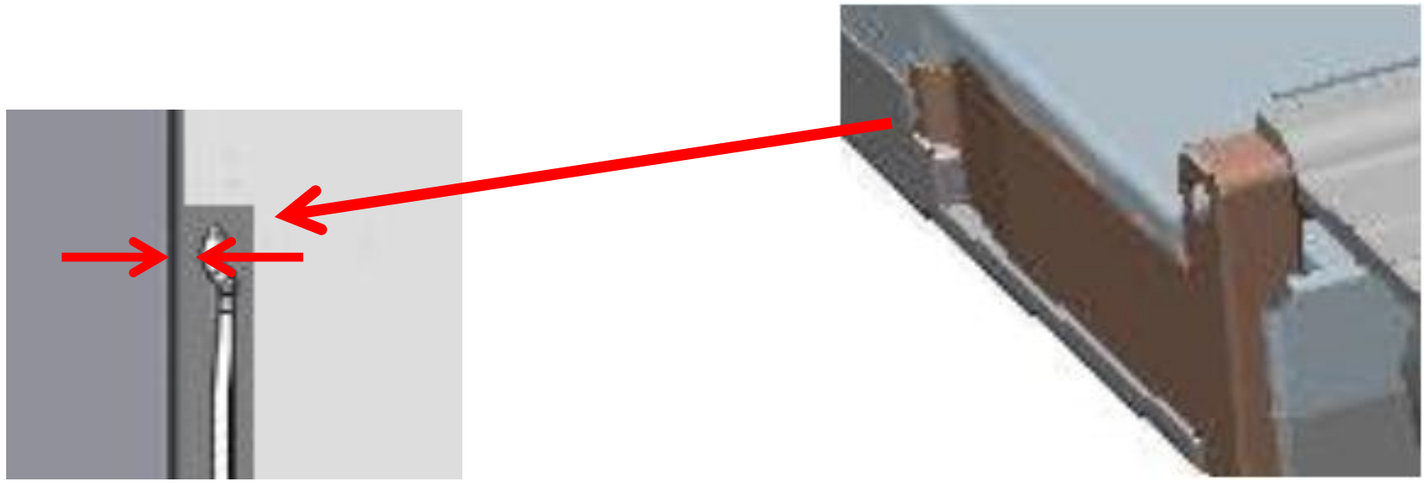
Introduction

- Specification for QSFP mechanicals and low-speed electricals is SFF-8436
- QLogic has tested numerous vendors' cable assemblies and has developed a list of common failure modes observed across multiple vendors
- This list is intended to help
  - New entrants into the QSFP cable market to be aware of past mistakes
  - Help current QSFP cable provides to avoid pitfalls in existing modules, due to manufacturing variation or future engineering changes
  - Drive QSFP spec improvement, in areas that are under-specified or ambiguously specified
- The failures listed in this presentation have been resolved in most if not all cases by the associated vendors (who will remain anonymous)
- Cables qualified by QLogic do not exhibit any of these failures
  - Refer to the QLogic website for a list of qualified assemblies

Mechanical pitfalls

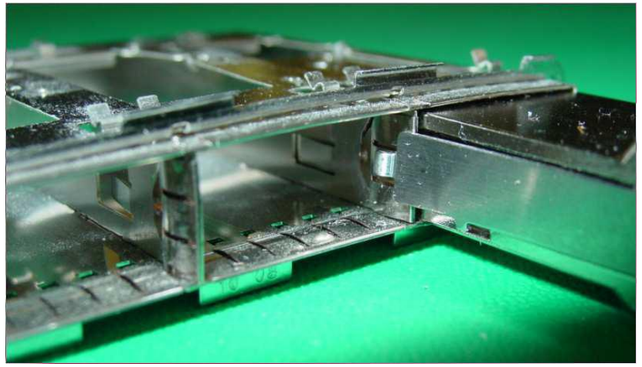
Mechanical – Cables get stuck in ports

- Most common ‘cable gets stuck’ root cause is improper delatching mechanism on the plug assembly
- If the delatching arm on the side of the module are too far recessed in the module shell, the cage latch is not fully disengaged and the cable cannot be removed without tools
- Failures we’ve seen are marginal (sometimes work on some cages)
- Spec may need more/better dimensions



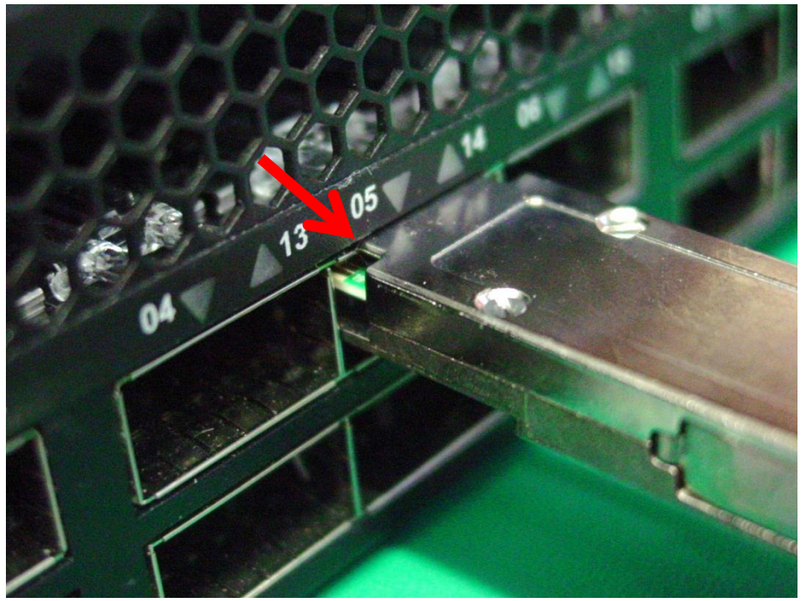
Mechanical – Cables get stuck in ports

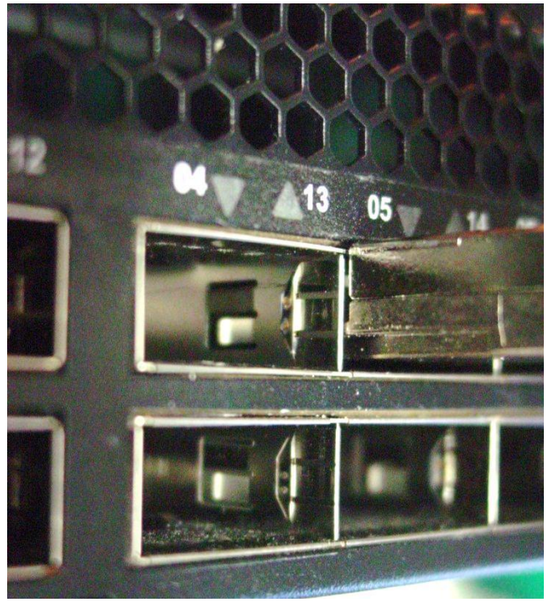
- ‘Cable gets stuck’ failure also due to delatch mechanism
- This failure can occur if the delatch arm on the module is outside the profile of the plug module shell
- When inserted into a ganged cage, the plug goes into its port but the delatch arm goes into the adjacent port, wedging the module
- Could be manufacturing-time escape
- Could be due to the rigid connection between the latch arm and the user insertion/extraction tab
  - Insertion/extraction tab is a natural push vehicle for installer
  - If cable is at all misaligned to the port, creating resistance, pushing on the tab will torque the latch arm assembly
  - This forces the latch arm tip outside of the profile of the module housing



Mechanical – Module stubbing damages

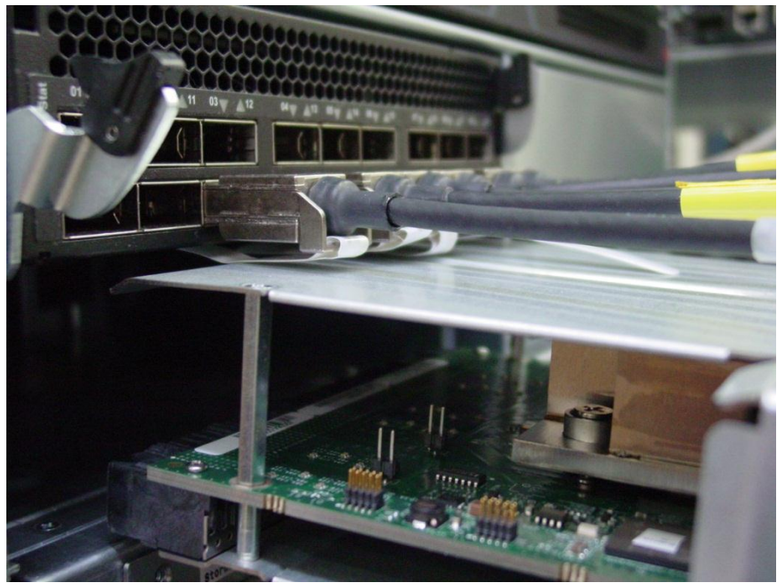
- Through-bezel cages can expose this problem
- If the QSFP plug stubs when inserted, components near the pads on the secondary side of the paddle card are susceptible to damage
- Should the secondary side keepout be increased?

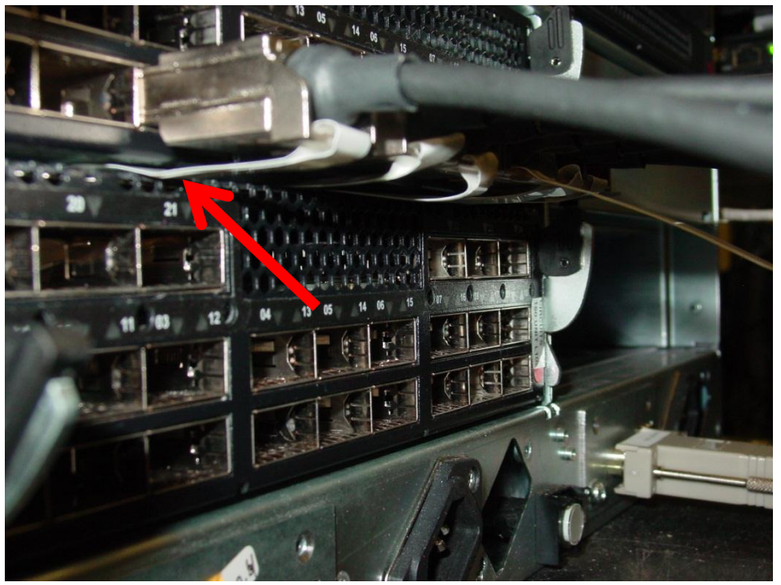




Mechanical – Pull tab gets stuck in adjacent card

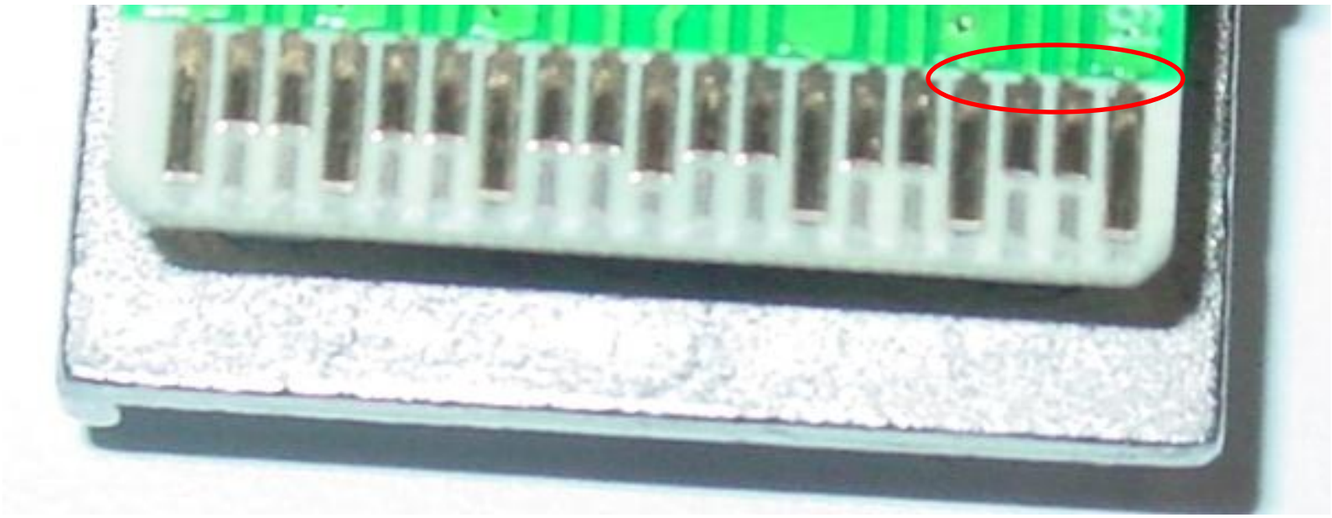
- If pull tab is free rotating, it can get stuck in the adjacent card slot
- Can occur with rigid or flexible pull tabs
- Flexible pull tabs also can get ‘rolled up’ in between modules





Mechanical – Pads on paddle card are too short

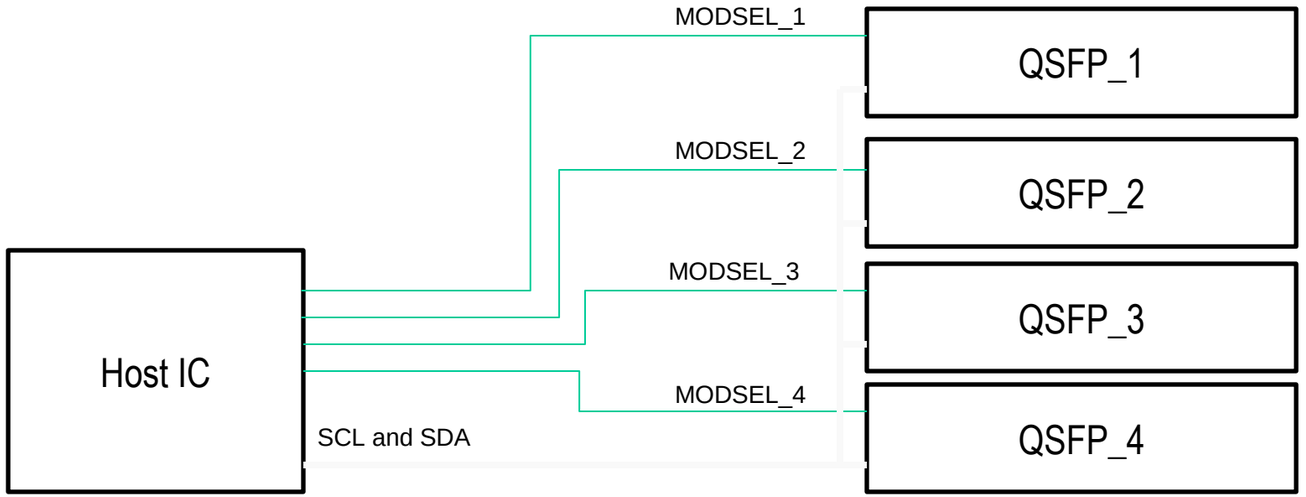
- If pad depth into paddle card is too shallow, contacts over-travel pads and make intermittent contact
- Check the wear tracks!
- Some QSFP receptacles and cages may work fine, others may not
- Beware of things like soldermask misregistration on the paddle card



I2C pitfalls

I2C bus – MODSEL not implemented properly

- Shared I2C bus uses MODSEL to identify which cable is being talked to
- A ‘bad’ cable will appear to read fine if it is the only device on the I2C bus
- However, a ‘bad’ cable will cause all other cables on the same I2C bus to stop reading, whether those other cables are good or bad
- A test is implemented for PlugFest to check for this, cable will fail if not implemented correctly



I2C bus – Clock stretching duration too long

- I2C rules dictate that a slave device (cables are slave devices) can hold the SCL line low if more time is needed to process an instruction
- If the line is held low too long, the I2C master (InfiniBand ASIC) may think the bus is stuck and cancel the transaction
- While the SCL line is held low, no other cables on the same I2C bus can be read

Table 12- QSFP+ Memory Specification

Parameter	Symbol	Min	Max	Unit	Conditions
Serial Interface Clock Holdoff "Clock Stretching"	T_clock_hold		500	us	Maximum time the QSFP+ Module may hold the SCL line low before continuing with a read or write operation

Memory map pitfalls

Memory map – Supported speeds byte

- Supported speeds byte 164 is used by devices to determine which speeds to attempt during link negotiation
- This byte is a bitmask, so ALL supported speeds must have a 1 for their associated bit in the bitmask
- Cables and devices are not required to implement intermediate speeds
  - SDR is always required; it is used for link negotiation
- InfiniBand endpoint ASICs are expected to query the QSFP cable supported speeds bitmask and do a logical AND with their own supported speeds bitmask before beginning link negotiation

Table 38 – Extended Module Code Values (Address 164) (Page 00)

Address	Bit	Description of Module Data
Infiniband Compliance codes		
164	7-5	Reserved
164	4	EDR
164	3	FDR
164	2	QDR
164	1	DDR
164	0	SDR

Memory map – Cable attenuation

- At DDR and QDR, cable attenuation is vital to link training
- For passive, linear active, and far end limiting active cables, attenuation tells the transmitter how much equalization to apply to the launched signal
- The next couple of slides define how to populate the attenuation field

186-187	2	Wave length or Copper cable Attenuation	Nominal laser wavelength (wavelength=value/20 in nm) or copper cable attenuation in dB at 2.5GHz (Adrs 186) and 5.0GHz (Adrs 187)	R	R
188-189	2	Wavelength tolerance	Guaranteed range of laser Wavelength (+/- value from nominal wavelength. (wavelength Tol.= value/200 in nm) 7.0 GHz (Adrs 188) and 12.9 GHz (Adrs 189)	R	R

Memory map – Cable attenuation, passive cables

- This value should be the attenuation measured at the indicated frequency, calibrating out the trace and test point loss of the module compliance board
  - Perform an end-of-cable calibration
  - Measure the SDD21 for the 2X THRU structure on the MCB
  - Measure the SDD21 for the lane of the cable under test
  - Subtract SDD21_cable_and_board – SDD21_2X_THRU = SDD21_cable_only
  - If the decimal is >= 0.5, round up to the nearest whole dB, otherwise round down
  - This process assumes a clean (low reflection/ringing) MCB is used
- This process is used for each speed's attenuation value
- Cable attenuation should be the nominal attenuation by design and does not need to be measured for every individual assembly at production-time
- This value is very important because it is used for accurate link tuning
- A common error is failing to remove the board loss

Memory map – Cable attenuation, active cables

- For full active copper and fiber as well as near end limiting active cables, the attenuation field in the cable is ignored
- For linear active and far-end limiting active cables, the attenuation field can be used to control the host transmitter equalization
- Currently, the specification does not define a more specific methodology for determining the attenuation value for these cables
- We are looking for a cable vendor to help us to develop the equalization methodology specification for linear active and far-end limiting active cables

Memory map – active cables faults and warnings

- QSFP provides a variety of warning and alarm monitors in the memory map
- Common errors in module monitors
  - VCC warnings
    - Warnings or alarms are reported when there should be no alarm
    - Often a result of the granularity of the monitoring circuitry in the module
  - TX fault
    - Don’t generate a TX fault when the other end of the cable gets unplugged
  - LOS
    - Should LOS ever be asserted for an AOC?
- These warnings do not result in any change in behavior by QLogic devices, but they do create log events that can be seen by the end user
- We should keep alarms and warnings to cases where there is actually a problem

Memory map - checksum

- Don’t forget to update the EEPROM checksum if you make changes to the memory map
- EEPROM checksum fields
  - CC_BASE
    - Stored in byte 191
    - Checksum for addresses 128-190
  - CC_EXT
    - Stored in byte 223
    - Checksum for addresses 192-222