ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

Single and Multi-Port 10/100M PHY POPI Requirements

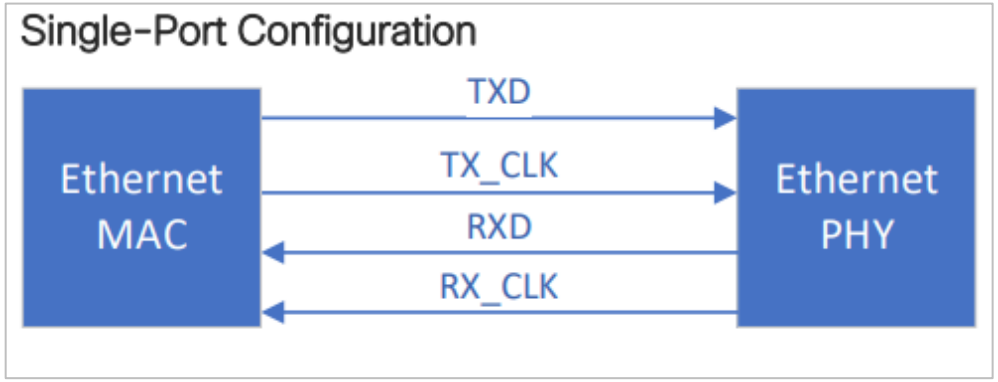
Brian Murray
Jacobo Riesco

Introduction

- This presentation is a continuation of our January and March presentations to develop the POPI requirements for a 10M and 100M single PHYs
  - With particular focus on the 802.3dg 100BASE-T1L PHY standard under development
  - Include the requirements for a full-duplex 10BASE-T1L PHY
  - And be compatible with legacy full duplex 100M PHY standards, e.g. 100BASE-TX and 100BASE-T1
  - Also consider the requirements for multi-port PHYs
- The requirements discussed include the following:
  - The pin interface requirements
  - Ordered sets for encapsulation and control signalling
    - Start and end of packet, carrier extend, error propagation, LPI, remote/local faults, PLCA and configuration
  - Interleave of PHY and sideband data streams / interleave of multi-port PHY and sideband data streams
  - Embedded sideband channel for the ingress and egress time stamps for PTP
  - Embedded sideband channel for MDIO and link information
- We should follow a consistent approach and logical format for POPI for lower and higher speed PHYs and for single and multi-port PHYs
  - The same frame formats for the embedded sideband channels

Review of POPI Pin Interface for a 100M Single PHY

- For a single 10M or 100M PHY the desire is to have a simple low pin count interface using standard IOs and digital logic
  - 100M data with 25% overhead for 8B/10B control/encapsulation
  - Add another 25% overhead for PTP time stamping and MDIO
  - Operate at a raw data rate on the pins of 156.25 Mb/s
  - Use single ended IOs to avoid the complexity of a SerDes while still keeping to a very low pin count
- Use a simple 4-pin interface with single ended IOs
  - TXD, TX_CLK, RXD, RX_CLK
  - Same pin count as a SerDes (4 pins)
  - Signalling rates are similar to existing GE PHY MAC interfaces (< 250 Mb/s per pin)
  - Clock/data recovery is optional
  - Source synchronous
  - Proven technology, low power/area, available in every technology node
    - An area efficient 1G SerDes might get close to the area of an RGMII MAC interface with 12 pins, but not 4 pins



Pin Optimized PHY Interface, Call For Interest Consensus Building (page 23)

Encapsulation and Control Signalling

- Use 8B/10B and Clause 36 ordered sets for control and encapsulation
  - This is a proven approach used in SGMII, QSGMII and USGMII
  - IDLE ordered sets are used for idle
  - Encapsulation ordered sets are used for the start and end of packet, carrier extend, error propagation
  - LPI ordered sets are used for Energy Efficient Ethernet
  - Configuration ordered sets
    - In clause 37 these convey information about the 1000BASE-X link
    - POPI could use Configuration ordered sets to configure the POPI link
    - Information about the PHY link should be in a sideband channel
- We will need to amend/supplement these ordered sets to include fault signalling and PLCA
  - Add Assert Local Fault and Assert Remote Fault from clause 46 and 190
  - Add PLCA Beacon Request and Commit Request from clause 22

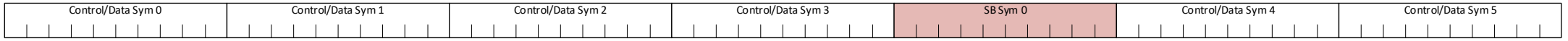
Table 36–3—Defined ordered sets

Code	Ordered Set	Number of Code-Groups	Encoding
/C/	**Configuration**		Alternating /C1/ and /C2/
/C1/	Configuration 1	4	/K28.5/D21.5/Config_Reg[a]
/C2/	Configuration 2	4	/K28.5/D2.2/Config_Reg[a]
/I/	**IDLE**		Correcting /I1/, Preserving /I2/
/I1/	IDLE 1	2	/K28.5/D5.6/
/I2/	IDLE 2	2	/K28.5/D16.2/
	Encapsulation		
/R/	Carrier_Extend	1	/K23.7/
/S/	Start_of_Packet	1	/K27.7/
/T/	End_of_Packet	1	/K29.7/
/V/	Error_Propagation	1	/K30.7/
/LI/	**LPI**		Correcting /LI1/, Preserving /LI2/
/LI1/	LPI 1	2	/K28.5/D6.5/
/LI2/	LPI 2	2	/K28.5/D26.4/

[a]Two data code-groups representing the Config_Reg value.

POPI for a Multi-port PHY – Interleave Ports & Sideband

- For a single-port PHY, POPI can interleave PHY and sideband 8B/10B symbols
  - An interleave ratio of 4:1 gives us 25 Mb/s for the embedded sideband channel
  - Run the POPI interface 25% times faster than the 8B/10B encoded payload rate

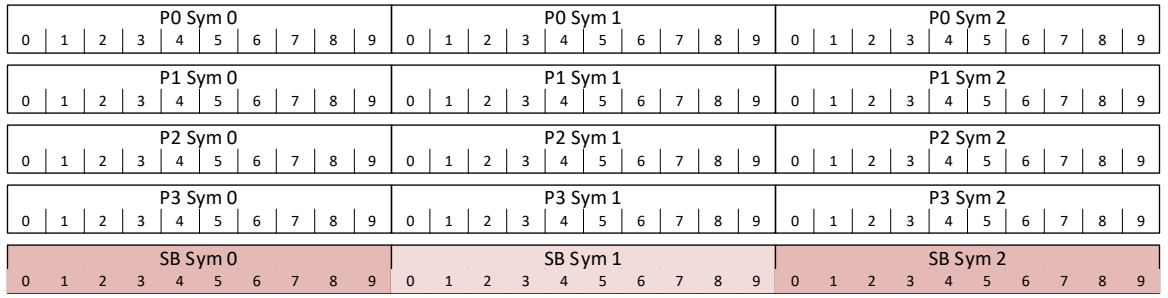


156.25 Mb/s

- For a multi-port PHY, POPI can share a sideband channel over a number of ports
  - So we can interleave a 100M sideband channel with the 100M PHY ports
  - A quad is a very common sub-multiple, so one obvious choice is to interleave a sideband every 4 PHY ports
  - This gives us 25 Mb/s for the embedded sideband channels for each PHY
  - Run the POPI interface 25% times faster than the multi-port 8B/10B encoded payload rate
    - The 4 data bytes are each encoded as 8B/10B and sent first, followed by the sideband byte as a 10B symbol

Bytes encoded 8B/10, and aligned

Sideband byte 8B/10B encoded



125 Mb/s
125 Mb/s
125 Mb/s
125 Mb/s
125 Mb/s

Ports are interleaved and 10B symbols sent at 5 x rate

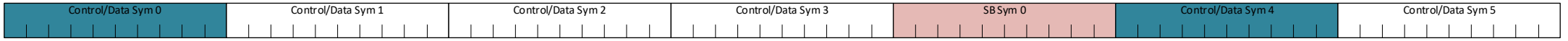
625 Mb/s

Identification of Sideband Symbols and Port Numbering

- In 8B/10B encoding COMMA codes allow easy alignment of the 8B/10B symbols within the serial stream
  - There are guaranteed bit transitions to support clock data recovery
- However, a means is required to distinguish the sideband symbols from data symbols
- In a multi-port PHY if we identify the symbol for Port 0 we also identify Port 1, Port 2, Port3 and the sideband symbol as they follow in sequence
  - In QSGMII and USGMII a scheme is used where they swap the K28.5 codes with K28.1 for Port 0
  - Hence any Idle symbol or other control symbol on Port 0 can be used to identify Port 0
  - The exact same scheme can be use for POPI for multi-port PHYs
- In a single PHY the sideband symbols are interleaved every 4 data symbols
  - We can use a similar scheme and swap the K28.5 codes with K28.1 for the first data symbol after the sideband symbol and every 4^{th} data symbol after that

POPI for a Multi-port PHY – Interleave Ports & Sideband

- For a single-port PHY, POPI can interleave PHY and sideband 8B/10B symbols
  - An interleave ratio of 4:1 gives us 25 Mb/s for the embedded sideband channel – run 25% faster than payload
  - For every 4th control/data symbol swap the K28.5 control codes with K28.1

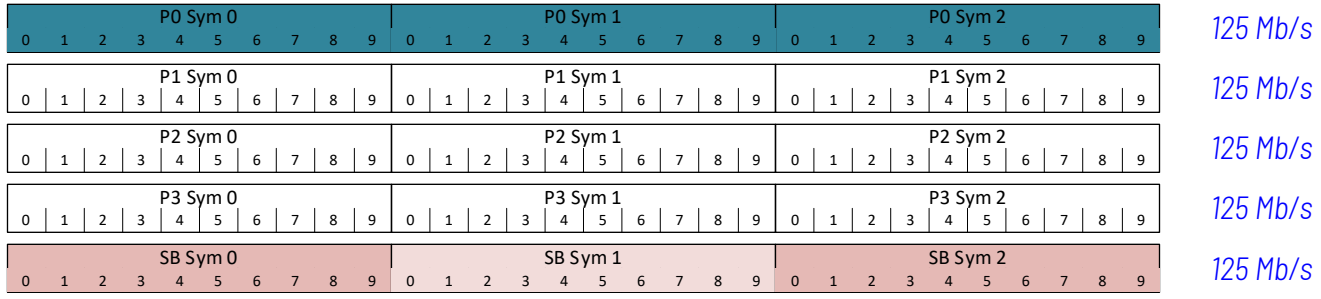


156.25 Mb/s

- For a multi-port PHY, POPI can share a sideband channel over a number of ports
  - So we can interleave a 100M sideband channel with the 100M PHY ports every 4 PHY ports
  - This gives us 25 Mb/s for the embedded sideband channels for each PHY
  - Run the POPI interface 25% times faster than the multi-port 8B/10B encoded payload rate
    - The 4 data bytes are each encoded as 8B/10B and sent first, followed by the sideband byte as a 10B symbol
  - For Port 0 control/data symbol swap the K28.5 control codes with K28.1

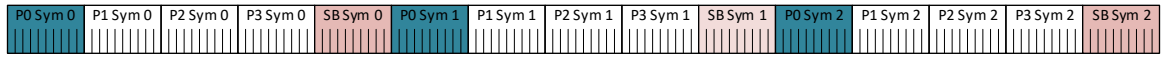
Bytes encoded 8B/10, and aligned
- Port 0 uses K28.1

Sideband byte 8B/10B encoded



125 Mb/s
125 Mb/s
125 Mb/s
125 Mb/s
125 Mb/s

Ports are interleaved and 10B symbols sent at 5 x rate



625 Mb/s

Review of Bandwidth Requirements for Sideband

- In previous presentations we have pre-viewed the bandwidth required for MDIO read and write and ingress and egress time stamps for PTP
  - Embedded sideband channel for MDIO could require **6.25 Mb/s**
    - Clause 22 specifies an interface speed of 2.5 MHz for MDIO, many PHYs support speeds up to 6.25 MHz
    - The 6.25 Mb/s already includes the overhead of management frame format – 16-bit payload is ~50% of frame
    - Note, that for a 10M PHY the MDIO bandwidth could be equal to 50% of the PHY data bandwidth
  - Embedded sideband channel for Ingress timestamps for PTP could require **12.5 Mb/s**
    - On the receive side a timestamp is usually sent from the PHY to the MAC for every packet
    - This covers the MACsec case where the payload is encrypted and the PTP packets cannot be identified
    - This would be a payload 4 byte timestamp for every 64 byte packet = 6.25 Mb/s
    - Allowing for 3 to 4 bytes of framing overhead this would require 12.5 Mb/s
  - Embedded sideband channel for Egress timestamps for PTP could require **2.5 Mb/s**
    - On the transmit side only some PTP packets require timestamp information to be sent from the PHY back to the MAC – but this could be a significant percentage of packets for some applications
    - Allow for 20% of transmit packets to be timestamped and worst case 64 byte packets = 1.25 Mb/s
    - Allowing for 3 to 4 bytes of framing overhead this would require 2.5 Mb/s
  - Also have some very low bandwidth link information; link status, speed, duplex, …

Use of Inter Packet Gap (IPG) for Ingress Timestamp

- You can see from the bandwidth requirements for MDIO and PTP that it is dominated by the Worst Case Ingress timestamp, ~60% of total bandwidth
  - In practice the bandwidth required for the Ingress timestamp for a typical case of a mix short and long packets would be much less; a lower percentage and lower absolute bandwidth

Channel	Peak and Nominal Rates	Nominal Bandwidth	Peak Bandwidth	Peak MDIO + Egress TS
MDIO	Peak = 6.25 MHz clock, all write or all read Nominal = 5 MHz clock, 40% write and 40% read	2.00 Mb/s	6.25 Mb/s	
Ingress TS	Peak = 4 bytes every 64 byte packet + 50% OH Nominal = 4 bytes every 512 byte packet + 50% OH	1.5625 Mb/s	12.5 Mb/s	
Egress TS	Peak = 4 bytes 20% of 64 byte packet + 50% OH Nominal = 4 bytes 5% of 512 byte packet + 50% OH	< 0.1 Mb/s	2.5 Mb/s	
		< 4.0 Mb/s	**21.25 Mb/s**	**8.75 Mb/s**

- One simple way to deal with the variation of the bandwidth required for the Ingress timestamp wrt the Rx packet length is to use the IPG
  - Guaranteed IPG of 12 bytes – use 5 to 8 of these 12 bytes for the Ingress timestamp + overhead
  - Ingress timestamp would be 5 data symbols following directly after the packet Terminate symbol
- This would reduce the Peak Bandwidth required and would allow a 1:8 interleave or at a 1:4 interleave an even higher data rate for MDIO or other sideband transactions

Peak and Nominal Rates for a 10M PHY

- A 10M PHY can operate a raw data rate on the pins of 15.625 Mb/s
  - The interleave ratio of 4:1 gives us 2.5 Mb/s for the embedded sideband channel
- For 10M, the bandwidth requirements for MDIO and PTP is dominated by the MDIO and Worst Case assumptions, ~80%of total bandwidth
  - In practice the bandwidth required for MDIO would be less for 10M links
  - But still looks marginal at a 25% overhead
  - A 10M PHY may need a 100% overhead for the sideband channel, which is a 1:1 interleave at 25 Mb/s
  - Or limit the MDIO bandwidth to 2.25 Mb/s

Channel	10M Peak and Nominal Rates	Nominal Bandwidth	Peak Bandwidth	Peak MDIO + Egress TS
MDIO	Peak = 6.25 MHz clock, all write or all read Nominal = 5 MHz clock, 40% write and 40% read	2.00 Mb/s	6.25 Mb/s	
Ingress TS	Peak = 4 bytes every 64 byte packet + 50% OH Nominal = 4 bytes every 512 byte packet + 50% OH	0.15625 Mb/s	1.25 Mb/s	
Egress TS	Peak = 4 bytes 20% of 64 byte packet + 50% OH Nominal = 4 bytes 5% of 512 byte packet + 50% OH	< 0.01 Mb/s	0.25 Mb/s	
		< 2.5 Mb/s	**7.75 Mb/s**	**6.5 Mb/s**

Summary

- For a single 10M or 100M PHY, POPI should be implemented as a simple 4-pin interface with single ended IOs and digital logic
- Should operate at signalling rates similar to existing GE PHY MAC interfaces, e.g. < 250 MHz
- Support MDIO, PTP and clause 36 like ordered sets as embedded interfaces with POPI to achieve the lowest pin count interface
- Swap K28.5 control codes with K28.1 to identify sideband symbols or Port 0
- Need to support 10 Mb/s to 25 Mb/s for the embedded sideband channels

Questions ?