ANALOG DEVICES

AHEAD OF WHAT'S POSSIBLE™

64B/66B Symbol Encoding for POPI for High Speed PHYs

Jacobo Riesco
Brian Murray

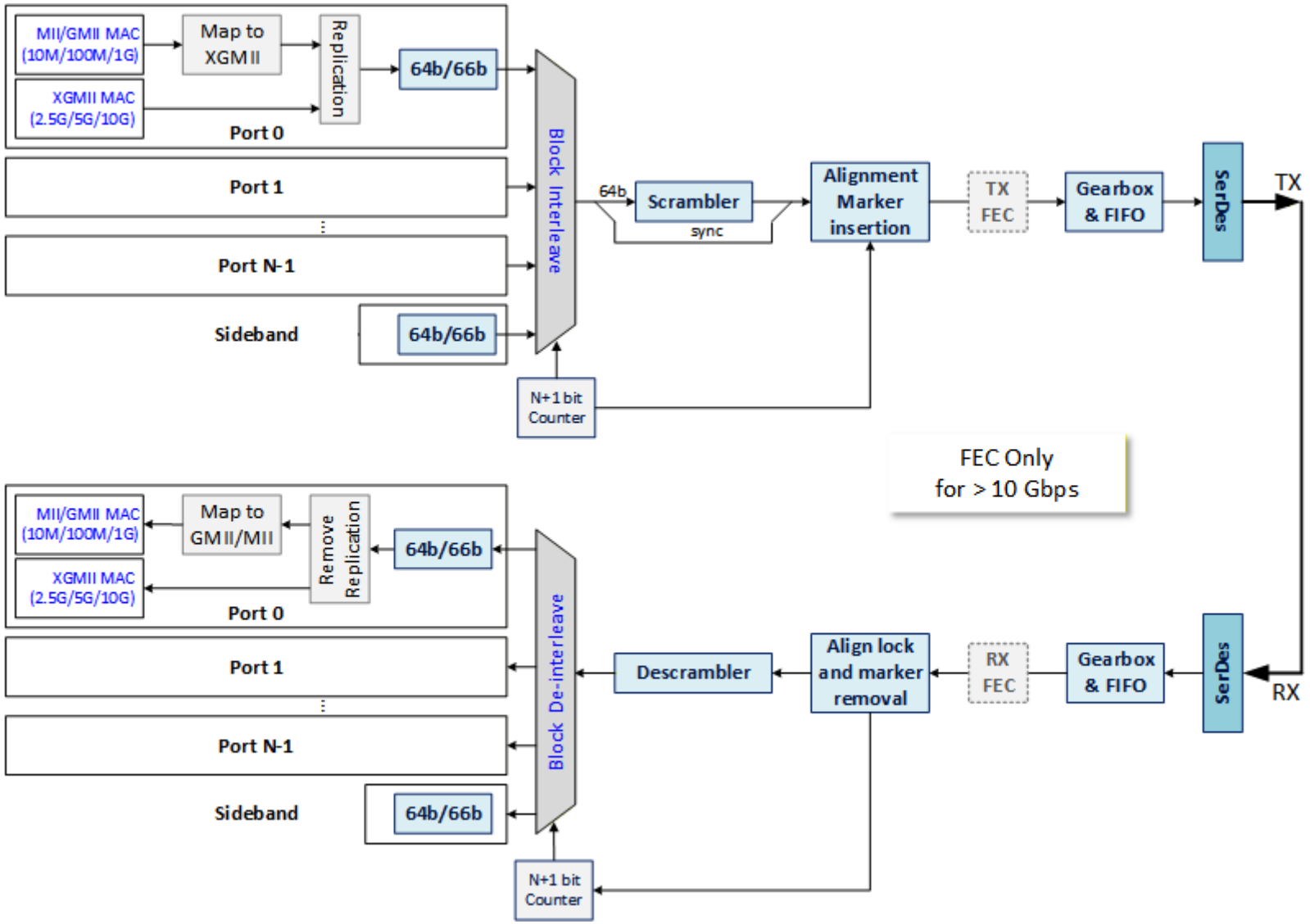
Introduction

- This presentation discusses some options for POPI for higher speed PHYs
  - This builds on previous presentations, for example see Mowad's presentation Multi-Port POPI Framework
- It is envisaged that POPI will operate using single ended digital IO for lower speeds (< 1G) and using SerDes for higher speeds (≥ 1G)
  - The lower and higher speed POPI will use a similar logical interface, keeping as much as is reasonable the same
- Build on past standards for higher speed PHYs and multi-port PHYs including QSGMII, USGMII, and USXGMII
  - Use 64B/66B encoding from 10GBASE-R in Clause 49
  - Source synchronous with clock/data recovery and scrambler for spectral integrity of the serial data
  - The single-port PHY interleaves the data 64B/66B blocks and sideband 64B/66B blocks in a 1 in 4 ratio
  - The multi-port PHY interleaves the port data 64B/66B blocks and has identification of Port 0
  - Support byte replication/stuffing for ports operating at lower speeds and for disabled ports
  - Have a sideband channel for embedded management including Ingress and Egress timestamps/TSSI
  - Support XGMII control codes and Sequence ordered sets
  - Exchange POPI link configuration at power-up and operate at lower speed when all the PHY links are down

64B/66B Encoding for Data Encapsulation

- Data encapsulation is necessary to support serialization of the data over a SerDes interface
  - The 64B/66B encoding provides encapsulation and regular synchronization information and include a multiplicative scrambler to ensure well behaved disparity and to ensure lots of transitions - this is achieved on a statistical basis
  - The same set of control characters are supported by the XGMII and the 10GBASE-R PCS
    - 10GBASE-R encodes the start and terminate control characters implicitly by the block type field
    - 10GBASE-R encodes the ordered set control codes using a combination of the block type field and a 4-bit O code for each ordered set
    - 10GBASE-R PCS encodes each of the other control characters into a 7-bit C code
    - The type fields are chosen with 4-bit Hamming distance
  - Scrambling is done after the encoding to ensure randomization of data and control signalling
  - The 2 sync bits are excluded from the scrambler to ensure fast block synchronization

Multi-port 64B/66B encoding



256b/257b Encoding with FEC at Rates > 10G

- For rates greater than 10G the PHY technologies 4 x 64B/66B blocks are transcoded as a 256B/257B block – clause 108
  - The sync bits are removed from the 64B/66B
  - 10GBASE-R and 25GBASE-R take twenty 256B/257B blocks and FEC added using an RS(528, 514) operating over the Galois Field GF(2^{10})
- There was a concern over the spectral spurs introduced by the sync bits in 64B/66B and FEC was considered desirable at the higher rates
- Higher speed POPI for a multi-port 2.5G/5G or single 10G PHY with 25% overhead for the sideband pushes us to rates > 10G
- **Do we need consider using 256b/257b with FEC for these cases**

64B/66B Encoding and Latency

- There is a much higher latency in bit times with 64B/66B encoding
  - Interleaving the 64B/66B blocks for each port and the sideband channel adds even more latency in bit times
  - However, at the much higher speed of operation (≥ 2.5Gbps) the latency in absolute time is acceptable and is consistent with current implementations of MAC-PHY interfaces

Scrambler/Descrambler

- IEEE Std 802.3 Clause 49 Physical Coding Sublayer (PCS) defines a *Self-Synchronizing (multiplicative) Scrambler* to *encode the 64 bit block* with polynomial: $G(x)=x^{58}+x^{39}+1$
  - No scrambler synchronization (training) required
  - When a single bit error occurs on the transmission link, it propagates through the LFSR and creates additional errors based on the number of feedback taps.
    - **Every single isolated bit error on the physical link results in exactly 3 bit errors** after descrambling. These three errors are spread across a span of **59 bits**
    - **May affect adjacent channels/sideband**
- An additive scrambler may be used instead to prevent error multiplication
  - Require scrambler synchronization/training
  - Used in other standards also using block encodings
    - PCIe Gen 3 through Gen 6+ (128b/130b and 242b/256b encoding) use a 23-bit additive scrambler with polynomial: $G(x)=x^{23}+x^{21}+x^{16}+x^{8}+x^{5}+x^{2}+1$
    - USB 3/Thunderbolt 3/Display Port 2.1 (128b/132b) use a 21-bit additive scrambler with polynomial: $G(x)=x^{21}+x^{19}+1$

Multi-port Alignment

- Multi-Port Alignment Markers (MP-AM) insertion/removal, borrowed from Code-Word Markers (CWM) in IEEE Std 802.3 clauses 82 and 108, are used as in USXGMII-MP for Port identification
  - Replace a 64b/66b block per channel periodically (every ~16K blocks in USXGMII-MP)
  - Idle (/I/) control characters, Low Power Idle control characters (/LI/), and ordered sets, are deleted to create room as necessary for the periodically occurring markers
- In POPI it may be simpler to use some special Sideband Idle Block Marker (SBM) periodically instead.
  - Identifies the side-band channel directly, instead of port 0, which follows
  - When the sideband channel is not sending a Macro Frame a control block with Idles is sent
    - This is 64B/66B block with a block type field 0x1E followed by eight 0x00 10GBASE-R control codes (/I/)
    - To distinguish the sideband channel from port data we could send eight 0x33 10GBASE-R control codes (reserved for K28.1)
  - In theory if you are sending continuous Macro Frames you should send an Idle block every ~16K
- CWM insertion/removal will still be required when RS-FEC is used at higher speeds
  - Note that the 10/25GBASE-R RS-FEC defined in Clause 108 uses 256b/257b transcoding which does not fit well with the 4 ports + 1 side-band channel architecture
  - This is an issue to be resolved if POPI is to operate at speeds higher than 10G

Summary

- A basic proposal has been presented using 64B/66B encoding with interleaved sideband channel for higher speed PHYs
- Build on past standards for higher speed PHYs and multi-port PHYs including QSGMII, USGMII, and USXGMII
- The approach used for the sideband channel and the linking and control information is closely aligned between lower and higher speed POPI
  - There are differences in the encoding and the interleave of the sideband channel that primarily related to achieving low latency for the lower speed PHYs
- Consideration of using an additive scrambler and alternatives for the multi-port alignment markers were discussed
  - These are aspects that may need to be addressed and should be considered further by experts in encoding for higher speed PHYs
  - What are the learning from 25G and above, do we an FEC?

Questions ?

Line Speed Spreadsheet (Updated) – Backup

Max Port Speed (Mbps)	# of Ports	Sideband Ratio	PCS Input Rate (Mbps)	Encoding	Encoding ratio	Line speed	Units	Sideband Effective Bandwidth	Sideband Bandwidth per Port
10	1	1.125	10	8b/9b	1.125	12.500	Mbps	1.25	1.25
100	1	1.125	100	8b/9b	1.125	125.000	Mbps	12.5	12.5
1000	1	1.125	1000	8b/9b	1.125	1.250	Gbps	125	125
1000	1	2	2000	8b/10b	1.25	2.500	Gbps	1000	1000
2500	1	1.25	3125	8b/10b	1.25	3.906	Gbps	625	625
2500	1	1.6	4000	8b/10b	1.25	5.000	Gbps	1500	1500
5000	1	1.25	6250	64b/66b	1.03125	6.445	Gbps	1250	1250
10000	1	1.25	12500	64b/66b	1.03125	12.891	Gbps	2500	2500
10	2	1.125	20	8b/9b	1.125	25.000	Mbps	2.5	1.25
100	2	1.125	200	8b/9b	1.125	250.000	Mbps	25	12.5
10	4	1.125	40	8b/9b	1.125	50.000	Mbps	5	1.25
100	4	1.125	400	8b/9b	1.125	500.000	Mbps	50	12.5
100	4	1.25	500	8b/10b	1.25	625.000	Mbps	100	25
1000	4	1.25	5000	8b/10b	1.25	6.250	Gbps	1000	250
2500	4	1.25	12500	64b/66b	1.03125	12.891	Gbps	2500	625
5000	4	1.25	25000	64b/66b	1.03125	25.781	Gbps	5000	1250
10000	4	1.25	50000	64b/66b	1.03125	51.563	Gbps	10000	2500
10	8	1.125	80	8b/9b	1.125	100.000	Mbps	10	1.25
100	8	1.125	800	8b/9b	1.125	1000.000	Mbps	100	12.5
100	8	1.25	1000	8b/10b	1.25	1.250	Gbps	200	25
1000	8	1.25	10000	8b/10b	1.25	12.500	Gbps	2000	250
2500	8	1.25	25000	64b/66b	1.03125	25.781	Gbps	5000	625
5000	8	1.25	50000	64b/66b	1.03125	51.563	Gbps	10000	1250
10000	8	1.25	100000	64b/66b	1.03125	103.125	Gbps	20000	2500