AQUANTIA

ACCELERATING CONNECTIVITY

EEE OAM Frame Proposal

Saied Benyamin (Aquantia)

Jim Graba (Broadcom)

Paul Langner (Aquantia)

Problem

- 802.3bp (1GBase-T1) defines an OAM frame with the following properties
  - Ten byte OAM data and 2 byte OAM CRC (total 12 byte frame)
  - In both data and LPI mode, the last payload byte of an RS frame carries the OAM
  - In LPI mode, the frame is only transmitted when it coincides with refresh and the data part of payload is replaced with zero’s
- 10GBase-T1 adds byte interleaving which complicates the matter. In order to get any information across the link (namely OAM), the minimum number of RS frames to transmit is equal to the interleave factor.
- This is not a problem in data mode, but in LPI mode, it would require a minimum refresh duration equivalent to the interleave factor.
- We propose a method for enabling us to send a single RS frame of refresh and preserve the ability to transmit OAM during LPI mode

Requirements

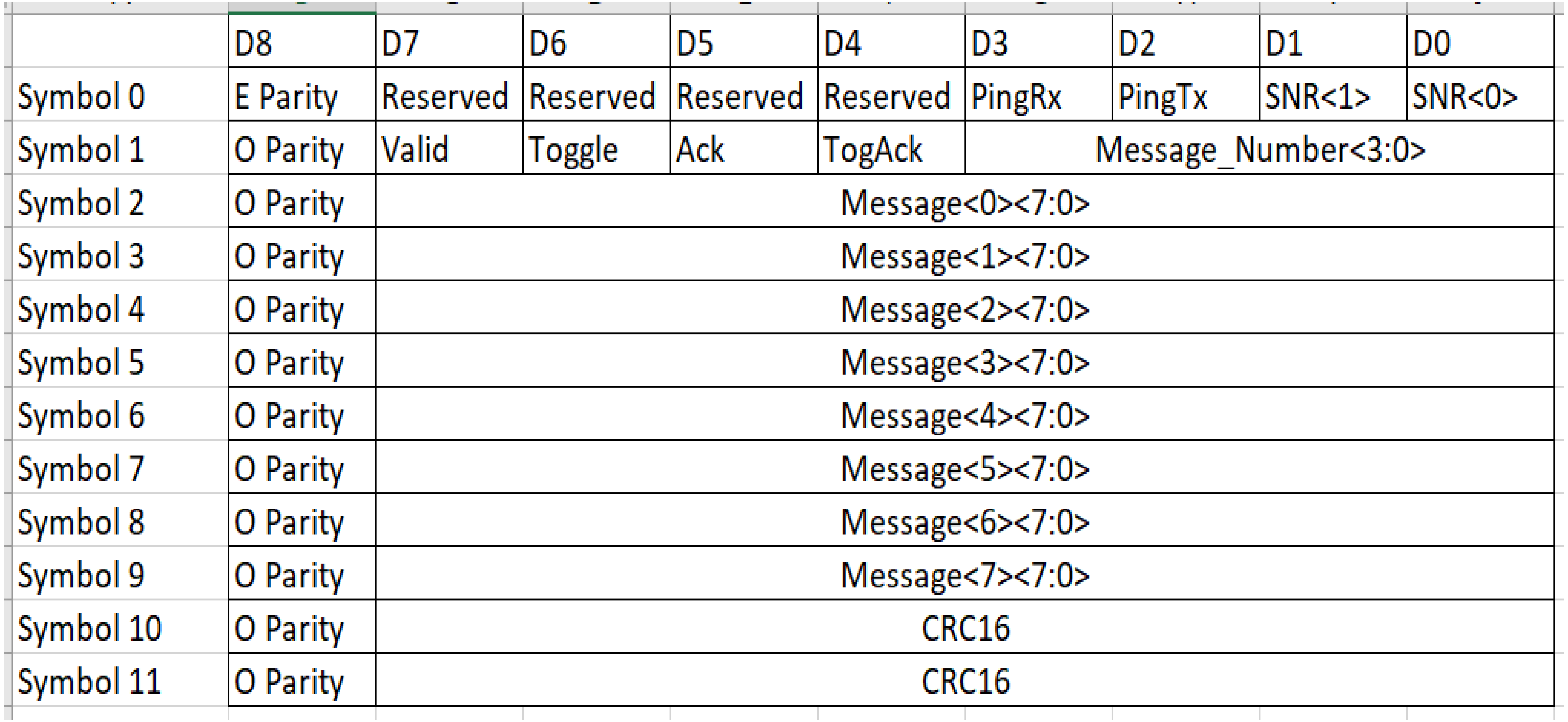
- Noise environment is low level Gaussian, with 60-110ns burst every 100us
- This means that 110ns of transmission data can be wiped out every 100uS
- We do not want to use interleaving during LPI refresh as it requires sending multiple RS frames per refresh, thereby extending refresh duration
- Following 1GBase-T1 method, a noise event could completely wipe out the OAM byte in the refresh which would corrupt the OAM frame

Solution

- Do not use the RS(360,326) Reed Solomon encoding to protect OAM during LPI
- Change current CRC error detection of OAM to Reed Solomon FEC RS(12,10)
- Only a single byte of the RS(12,10) encoded OAM frame is transmitted every refresh time as the last byte of refresh transmission.
- This forces the RS protected bytes far enough apart so that the 110ns of corrupted data can not affect more than one byte of OAM encoded RS frame.
- In this manner, an OAM byte wiped out by noise can be error corrected.
- Odds of two EMI events falling on refresh in a real life situation is less than 1:10,000.

OAM Frame – 1GBase-T1

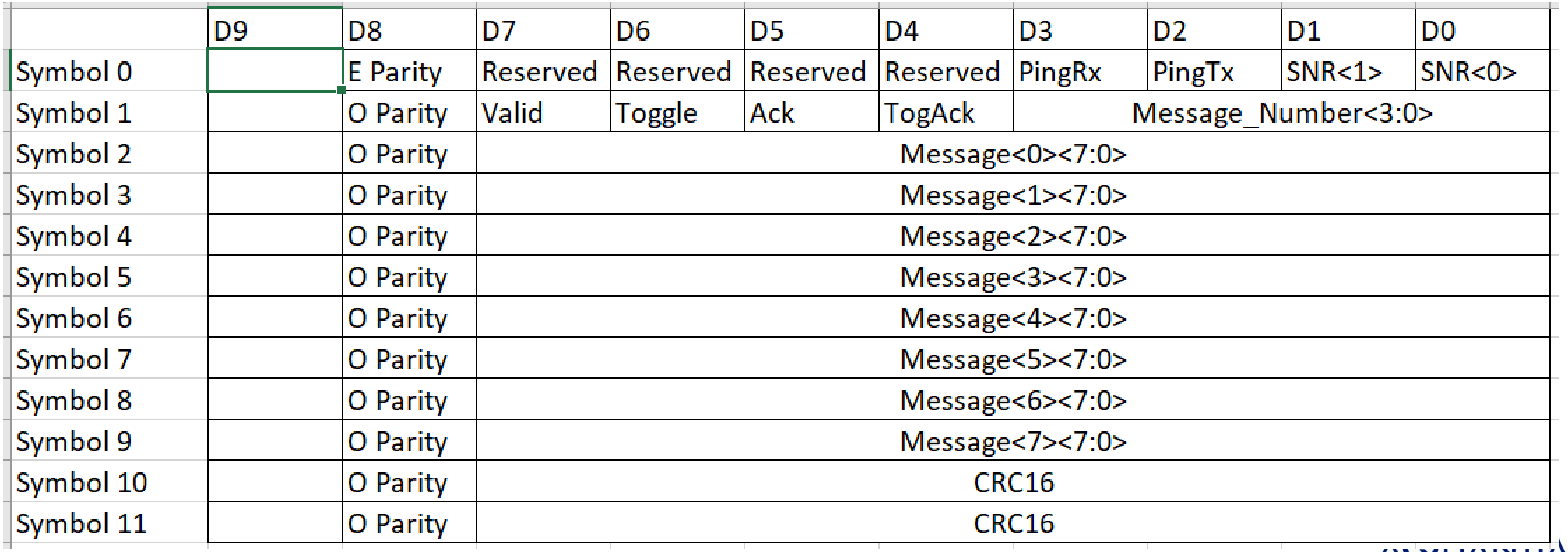
- The existing 1GBase-T1 OAM frame is depicted below



	D8	D7	D6	D5	D4	D3	D2	D1	D0
Symbol 0	E Parity	Reserved	Reserved	Reserved	Reserved	PingRx	PingTx	SNR<1>	SNR<0>
Symbol 1	O Parity	Valid	Toggle	Ack	TogAck	Message_Number<3:0>			
Symbol 2	O Parity	Message<0><7:0>							
Symbol 3	O Parity	Message<1><7:0>							
Symbol 4	O Parity	Message<2><7:0>							
Symbol 5	O Parity	Message<3><7:0>							
Symbol 6	O Parity	Message<4><7:0>							
Symbol 7	O Parity	Message<5><7:0>							
Symbol 8	O Parity	Message<6><7:0>							
Symbol 9	O Parity	Message<7><7:0>							
Symbol 10	O Parity	CRC16							
Symbol 11	O Parity	CRC16							

OAM Frame Proposal

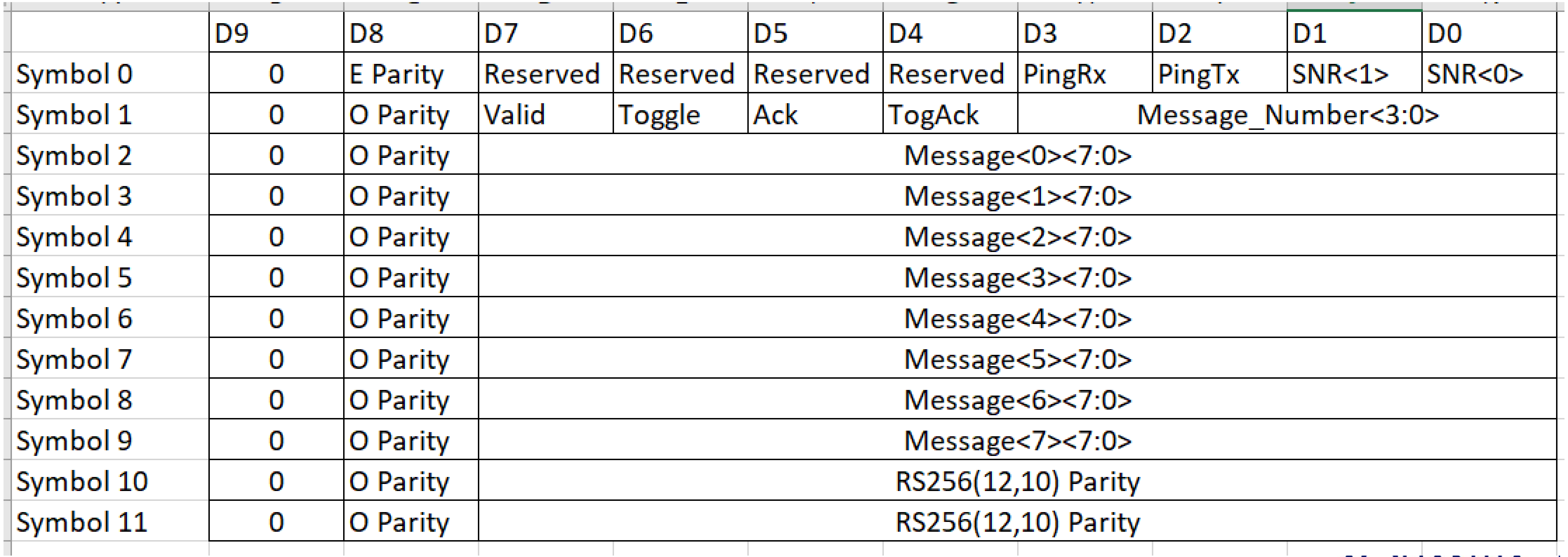
- 10GBase-T1 uses a 2^10 Galois field, whereas 1GBase-T1 used 2^9. So the OAM words need to be extended to 10 bits



	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Symbol 0		E Parity	Reserved	Reserved	Reserved	Reserved	PingRx	PingTx	SNR<1>	SNR<0>
Symbol 1		O Parity	Valid	Toggle	Ack	TogAck	Message_Number<3:0>			
Symbol 2		O Parity	Message<0><7:0>							
Symbol 3		O Parity	Message<1><7:0>							
Symbol 4		O Parity	Message<2><7:0>							
Symbol 5		O Parity	Message<3><7:0>							
Symbol 6		O Parity	Message<4><7:0>							
Symbol 7		O Parity	Message<5><7:0>							
Symbol 8		O Parity	Message<6><7:0>							
Symbol 9		O Parity	Message<7><7:0>							
Symbol 10		O Parity	CRC16							
Symbol 11		O Parity	CRC16							

OAM Frame Proposal – Alternate 1 – our choice

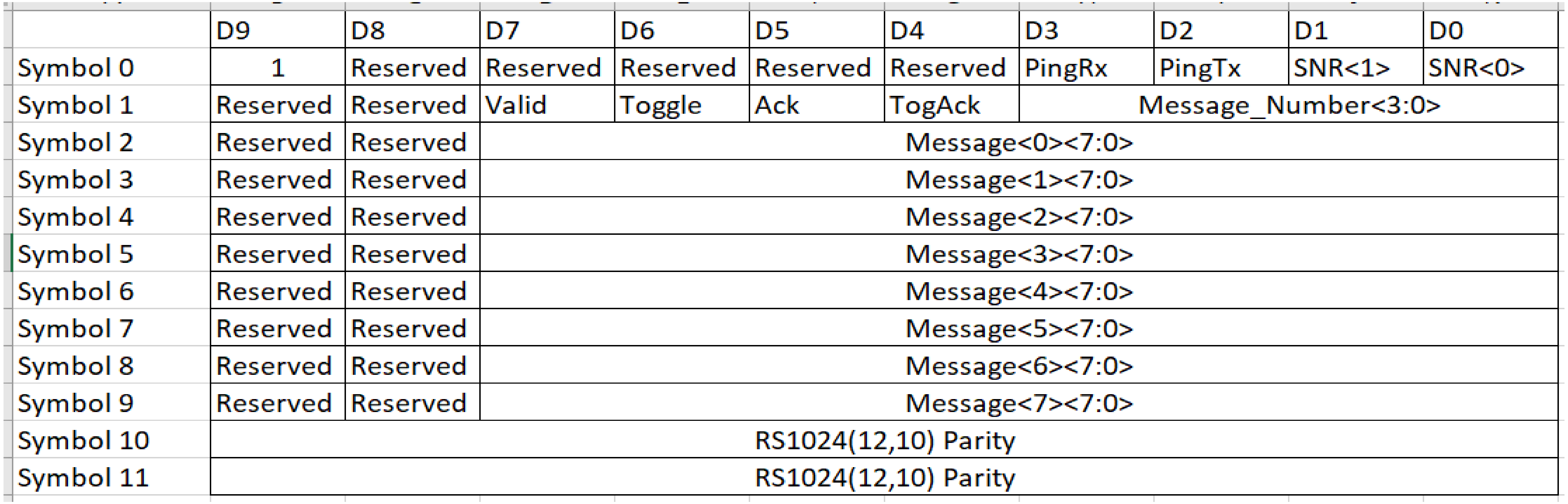
- Zero fill the new bits; replace the CRC bytes with an RS(12,10) parity code.
- LPI mode: new RS provides protection as well as frame boundary detection
- Data mode: can be used for frame boundary detection consistent with LPI mode or use even/odd parity alternately



	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Symbol 0	0	E Parity	Reserved	Reserved	Reserved	Reserved	PingRx	PingTx	SNR<1>	SNR<0>
Symbol 1	0	O Parity	Valid	Toggle	Ack	TogAck	Message_Number<3:0>			
Symbol 2	0	O Parity	Message<0><7:0>							
Symbol 3	0	O Parity	Message<1><7:0>							
Symbol 4	0	O Parity	Message<2><7:0>							
Symbol 5	0	O Parity	Message<3><7:0>							
Symbol 6	0	O Parity	Message<4><7:0>							
Symbol 7	0	O Parity	Message<5><7:0>							
Symbol 8	0	O Parity	Message<6><7:0>							
Symbol 9	0	O Parity	Message<7><7:0>							
Symbol 10	0	O Parity	RS256(12,10) Parity							
Symbol 11	0	O Parity	RS256(12,10) Parity							

OAM Frame Proposal – alternate 2

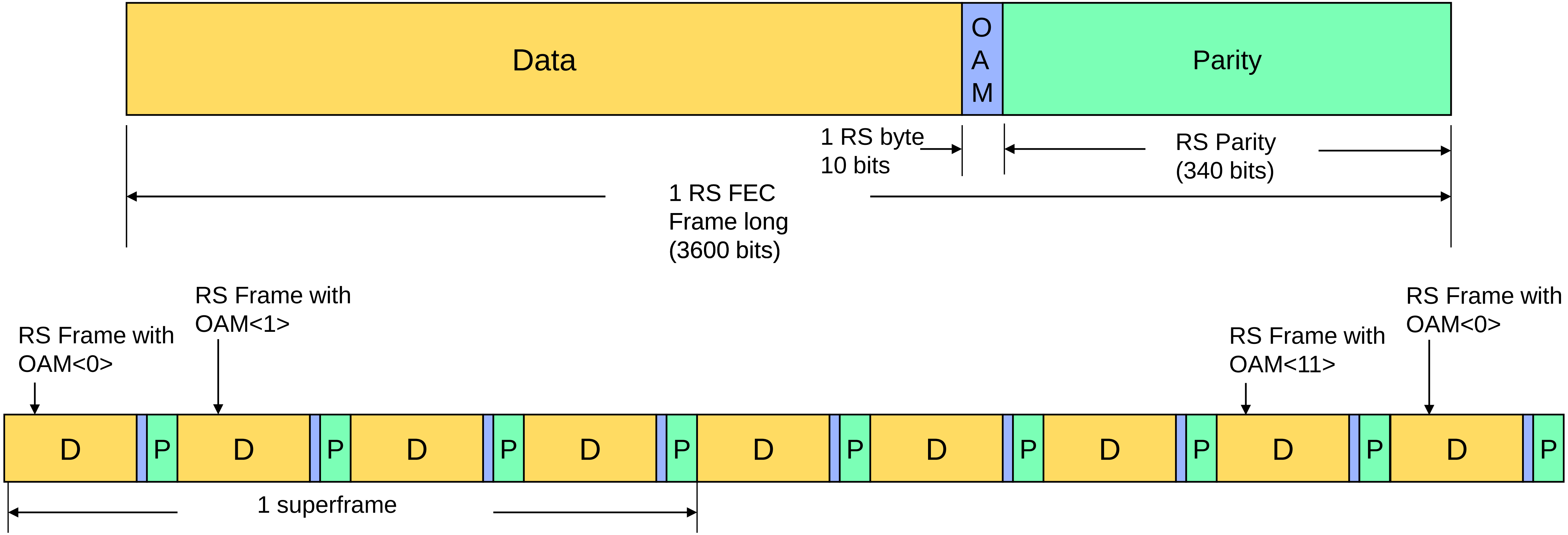
- We will mark the new bits and all the parity bits as reserved except one bit to assure that frame is not all 0’s which defeats the field alignment detection. The MDIO register definition changes will be added once accepted by members.
- Data protection: LPI uses RS(12,10); data mode already protected RS(360,326)
- Frame boundary detection: LPI and data mode use reed Solomon to detect the frame boundary.



	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Symbol 0	1	Reserved	Reserved	Reserved	Reserved	Reserved	PingRx	PingTx	SNR<1>	SNR<0>
Symbol 1	Reserved	Reserved	Valid	Toggle	Ack	TogAck	Message_Number<3:0>			
Symbol 2	Reserved	Reserved	Message<0><7:0>							
Symbol 3	Reserved	Reserved	Message<1><7:0>							
Symbol 4	Reserved	Reserved	Message<2><7:0>							
Symbol 5	Reserved	Reserved	Message<3><7:0>							
Symbol 6	Reserved	Reserved	Message<4><7:0>							
Symbol 7	Reserved	Reserved	Message<5><7:0>							
Symbol 8	Reserved	Reserved	Message<6><7:0>							
Symbol 9	Reserved	Reserved	Message<7><7:0>							
Symbol 10	RS1024(12,10) Parity									
Symbol 11	RS1024(12,10) Parity									

OAM during data mode

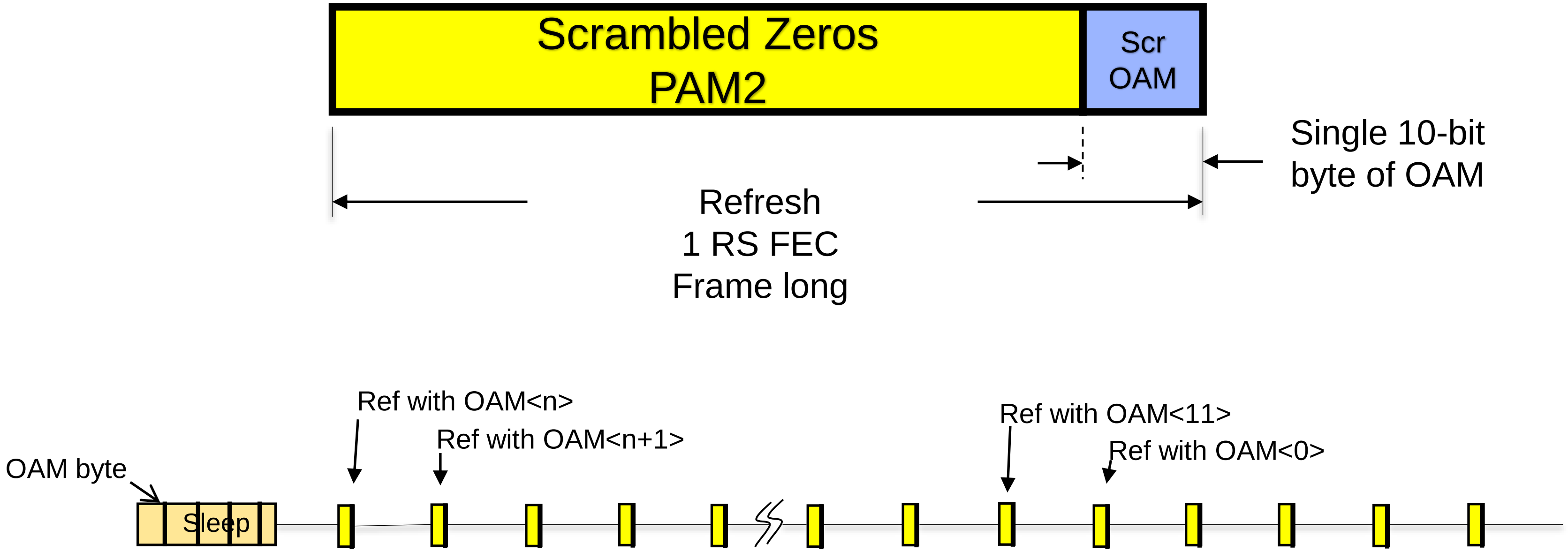
- In data mode, the resulting RS encoded OAM<11:0> is transmitted one byte per RS frame (similar to 1GBase-T1). OAM rate: 12 bytes per 3.8uS.



AQUANTIA

OAM during LPI

- The OAM is an RS(12,10) packet of data which is 12 bytes long. During LPI, a single byte is xor'd with scrambler bits being sent during refresh.
- OAM rate: 12 bytes per 480uS



Thank you.

AQUANTIA®

ACCELERATING CONNECTIVITY