IEEE 802.3br
Interspersing express traffic (IET)
Task Force (TF)

Baseline

2014-05-14

Contents

- Objectives Review
- MAC Merge Architecture
- Mframe format

Objectives Review

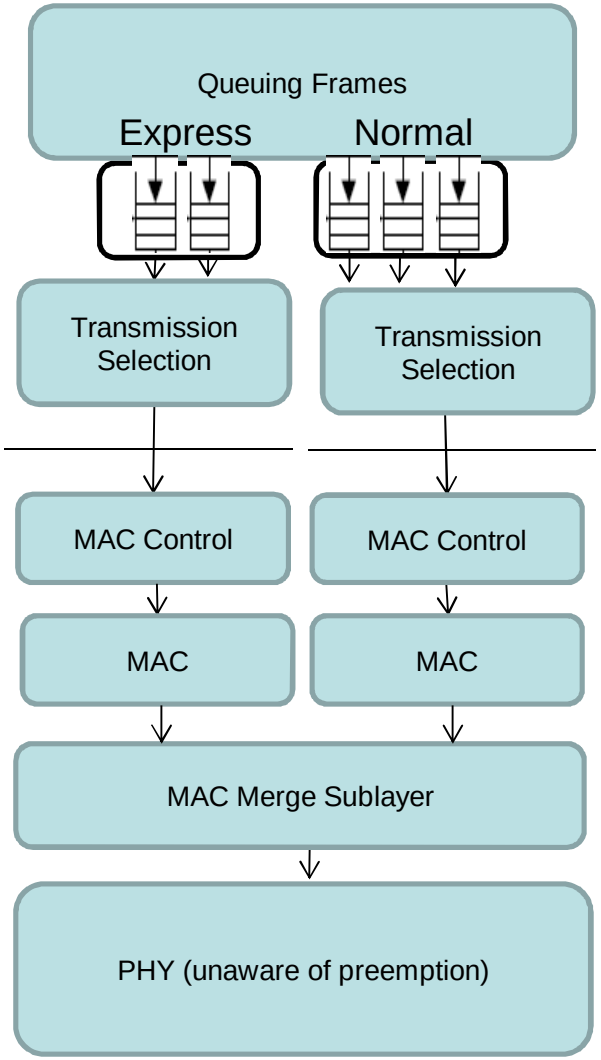
- ✓ 1. Preserve the IEEE 802.3 Ethernet frame format at the MAC.
- ✓ 2. Preserve minimum and maximum MAC frame size of the current IEEE 802.3 standard.
- ✓ 3. Use the Clause 4/Annex 4A MAC without alteration.
- ✓ 4. Require no changes to PHYs.
- ✓ 5. Support full duplex operation only.
- ✓ 6. Preserve MAC/PLS service interface.
- ✓ 7. Do not degrade Mean Time to False Packet Acceptance (MTTFPA) at the MAC Service Interface.
- ✓ 8. The latency to initiate the transmission of an express frame shall be less than two times the minimum packet size plus IPG.

Objectives Review

- ✓ 9. Assure that both ends of the link support Interspersing Express Traffic (IET) mode before enabling it.
- ✓ 10. Provide a primitive at the MAC client service interface to inhibit the transmission of non-express frames.
- ✓ 11. Provide two MAC client service interfaces at each end of the IET link, as the means to distinguish between the express and the non-express frames.
- ✓ 12. Minimum IET frame size shall be greater than or equal to 64 bytes.
- ✓ 13. IET frames will be constructed such that they will not be recognized as valid MAC frames by a non-IET-capable device.

MAC Merge layer encapsulation goals

- Preserve frame integrity
  - No increase in undetected errors
- Indicate which MAC receive frame belongs to
- Minimize impact on throughput
- Transparent to existing non-deprecated PHYs



Terminology

- Express frame – frames with the lowest latency
- Normal frame – frames that are not Express frames.
- Mframe -- A transmitted unit from MAC Merge that includes both whole frames and fragments of premptable frames – Mframe
    - Which stands for MAC Merge frame – a unit that looks like a frame at the PHY layer but may contain a whole frame or a fragment of a MAC layer premptable frame.

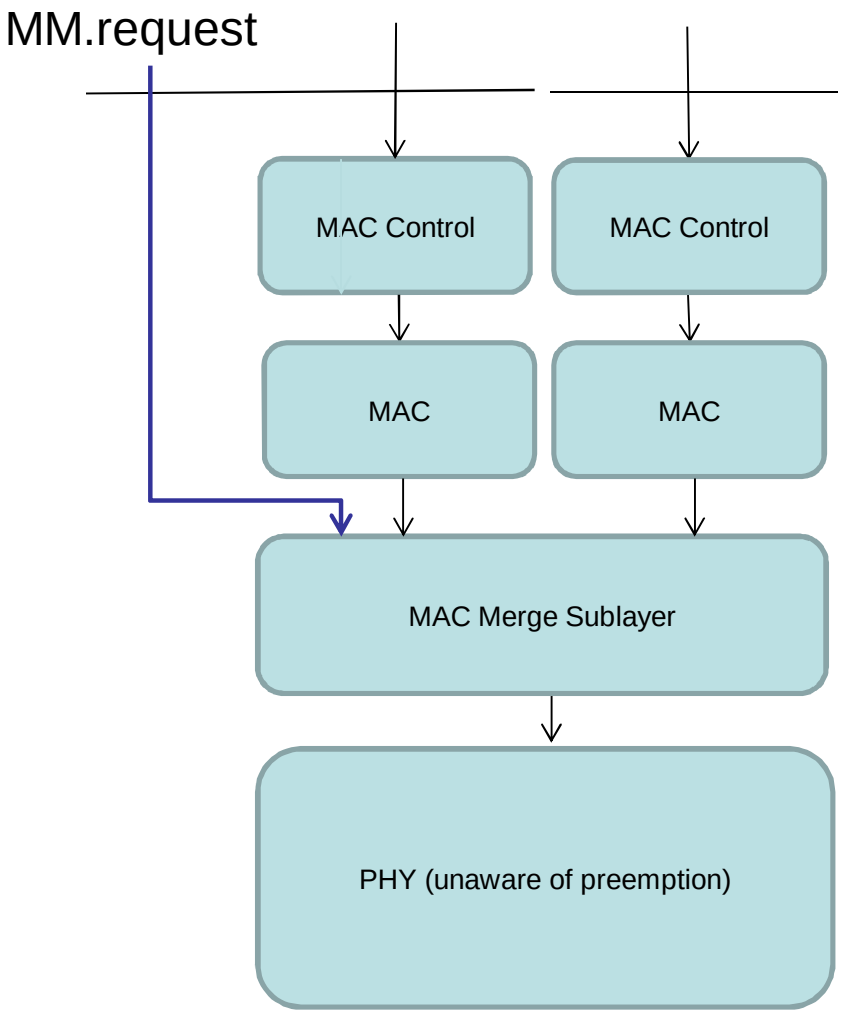
Hold Primitive need

A hold primitive allows:

- A MAC Client that has a schedule for Express traffic to preempt Normal frames before the scheduled Express traffic arrives When scheduled frame arrives, it can be transmitted immediately
- EPON MAC Control to preempt a frame near the end of the Gate and send a Report before the end of the Gate.

MM.Request from MAC Client

- MM.request: Primitive to carry the indication to Hold (i.e. preempt any frame in transmission and prevent the start of new frames) or release the normal transmission path

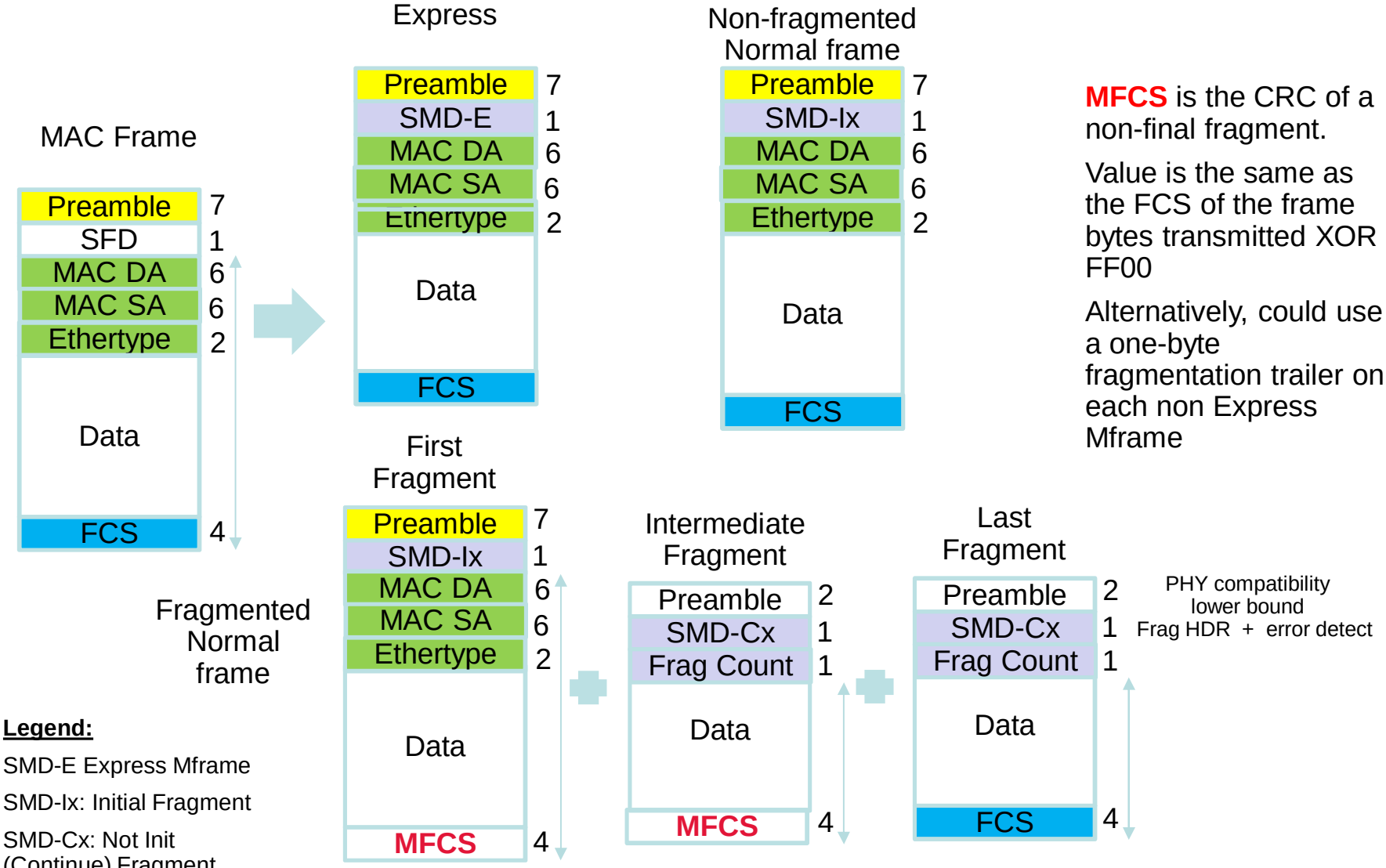


MM.request(hold_req)

- hold_req parameter takes one of two values:
  - hold – asserts hold variable in MAC Merge sublayer
  - release – clears hold value in MAC Merge sublayer
- MAC Merge preempts whenever
  hold = TRUE or
  IET MAC PLS_DATA.request has a bit to transmit.

Mframe format

Mframe format



MFCS is the CRC of a non-final fragment.

Value is the same as the FCS of the frame bytes transmitted XOR FF00

Alternatively, could use a one-byte fragmentation trailer on each non Express Mframe

Mframe format elements

- Preamble
  - Minimize use of preamble bytes for non-initial fragments
  - Provide at least 2 bytes of preamble for fragments
    - The 1000BASE-X PHY can drop up to 2 bytes of preamble and insert the SFD over another byte. In practice, implementations only drop 1 and many delay the start of preamble rather than drop and only overwrite 1 byte (similar to 10GBASE-X alignment)
    - Open issue: use full or shortened preamble for 10 Mbps PHYs?
- Identify Mframe as Express or start of a Normal frame or a later fragment of a Normal frame
- Protection for reassembly errors when an Mframe is lost
  - Frame number – circular count from 0 to 3
  - Fragment number – circular count from 0 to 3
- Identify last Mframe of a Frame
  - Mark end of Normal Mframe

Fragment size constraints

- Preempted fragment size will be no smaller than 64 bytes
  - Therefore a packet less than 127 bytes will not be preempted.
- To simplify implementation, non-final fragments will have an alignment constraint
  - That is, their payload will be a multiple of a set number of octets
  - For initial draft, leave TBD the alignment size as some multiple of 8 octets between 8 and 64 octets

Mframe start

- For start of non-initial fragments
  - Insert 2 bytes of preamble followed by
  - SMD byte (Start Mframe Delimiter)
  - Frag byte (Fragment count)
- Normal Frame start and Express frame
  - Replace SFD with SMD
- SMD values have Hamming distance 4 from each other
- Frag values have Hamming distance 4 from each other

SMD and Count byte encodings

Mframe type	Frame #	SMD
SMD-E	NA	0x55
SMD-Ix Premptable frame start	0	0x66
	1	0xCC
	2	0xFF
	3	0x33
SMD-Cx Non-initial fragment	0	0xE1
	1	0xD2
	2	0x1E
	3	0x2D

Frag Count	Frag
0	0x66
1	0xCC
2	0xFF
3	0x33

To comply with objective 13, 0xAA isn't used
Objective 13: IET frames will be constructed such that they will not be recognized as valid MAC frames by a non-IET-capable

Frame and Mframe CRC

- Frame CRC is generated by the MAC over the same frame bits as always. It is not altered by the MAC Merge sublayer

- MAC Merge layer calculates the Mframe CRC which it adds to any non-final fragment Mframes. Mframe CRC on a fragment is calculated over the bytes of the frame transmitted up to the end of the current Mframe. It is the same as the MAC CRC calculation XOR'ed with FF00

Fun with CRCs

- IEEE 802.3 inverts the calculated CRC and appends the result to the frame.
- Transmitting MAC Merge sublayer could invert the second two bytes of the intermediate CRC result (the CRC computed over the bytes of the MAC frame that have been transmitted so far) at the end of non-final fragments.
- Receiving MAC Merge sublayer runs a CRC calculation as the frame is received. When an Mframe ends, it compares the calculated value with the
  - If the difference between that and the last 4 bytes of the Mframe is 0xFFFF, it's the end of a MAC frame
  - If the difference between that and the last 4 bytes of the Mframe is 0x00FF, it's a non-final fragment.

Negotiation for IET

Assumptions for IET negotiation

- Negotiation will operate identically in each direction
  - Therefore – only 1 instance will be described
  - Standard will describe Tx & Rx behavior separately for both LP
- IET capable PHY will understand IET frames/fragments
  - No need to enable Rx – only Tx
- “Bad” device inserted between LP’s is unusual error
  - Mis-configuration can cause link failure after timeout
  - Bad packets will not be propagated
- Rapid startup must be possible but not mandated
  - Fast/slow startup left to implementer (application requirements)

LLDP - advertisement

- Each link partner sends an IET TLV:
  - One bit that indicates IET frames have been received
  - i.e. >0 IET frames received since last TLV
- System shall not transmit using IET frames…
  - … unless it is receiving IET TLV
- Simple 2-way advertisement
  - No state exchange needed

- IET frames received bit allows timeout for fault detection

Fault detection

- In scenario with invalid configuration
  - (illegal intermediate device)
- If all IET frames appear as invalid Ethernet frames
  - LLDP keepalives will fail
  - IET TLV will timeout – IET will cease
- In scenario where data is passed but SFD/preamble is overwritten (i.e. re-generated)
  - LP will not receive IET frames, IET received will not be returned
  - System may detect fault and act accordingly
- Fault detection will rely on timing of monitor/keepalive
  - Rare failure mechanism, not expected in most situations

Motion

To use this proposal as a baseline for the architecture and Mframe format in the draft.

Yes:

No:

Abstain

Moved: Albert Tretter

Second: Pat Thaler

Unanimously approved.