MPCP directed fragment reception

How to fragment without overhead

www.huawei.com

Author/ Email: Duane Remein

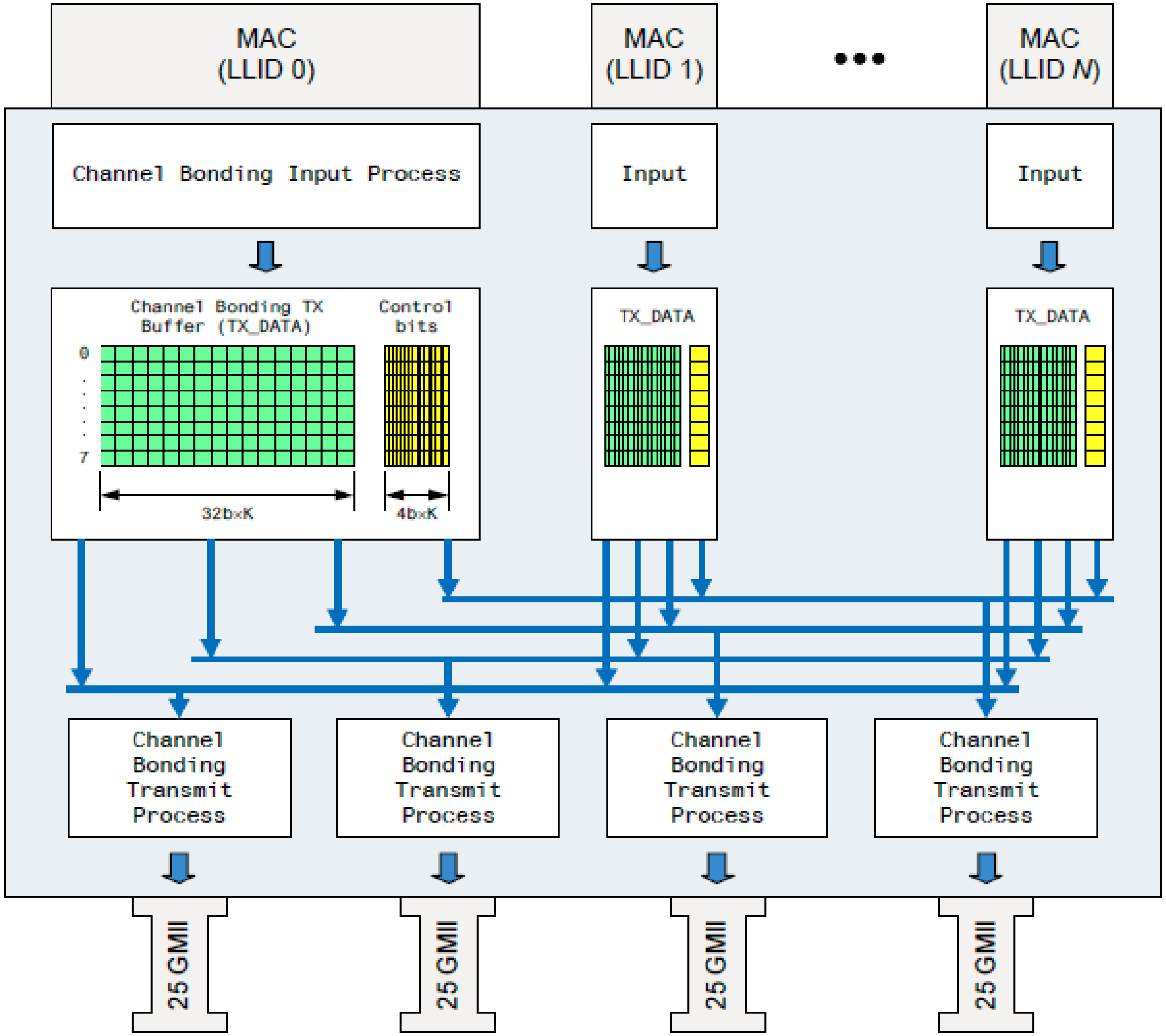
Version: V1.0(20160913)

HUAWEI TECHNOLOGIES CO., LTD.

HUAWEI

Idea adopted in San Diago

- To date all proposals have looked at fragmentation in the transmitter using markings to notify the OLT of fragment boundaries

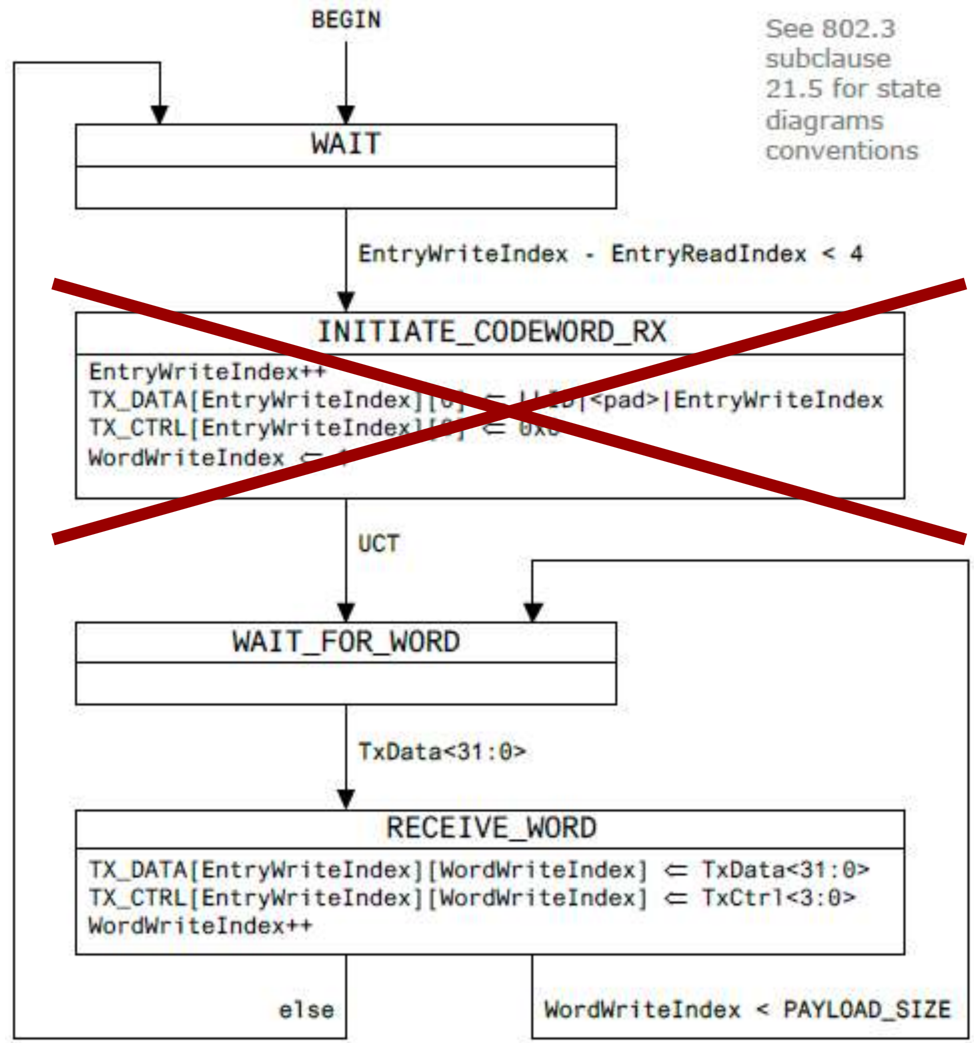


Can we avoid the markers?

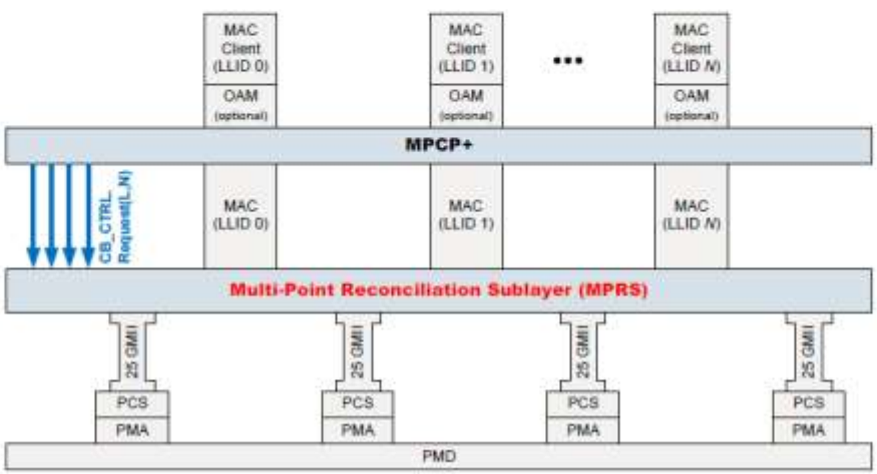
- **The MPCP layer in the OLT generated the grants so it must know where the fragment boundaries are.**
  - The Grant specifies a start time
  - The Grant specifies the duration
  - The Grant specifies the LLID
  - The Grant will need to specify the channel (xMII)
- **What if these were communicated to the receiving RS?**
  - RS would know when data is to arrive
  - RS would know how much data is expected (may include some IDLEs)
  - RS would know to which LLID the data is destined for
  - RS would know which channel the data is coming from

ONU Transmitter

- **Same as propose but no need to add markers**
- **Overhead reduction by ~1.8% (assuming 216 bytes per FEC codeword and 4 byte marker)**

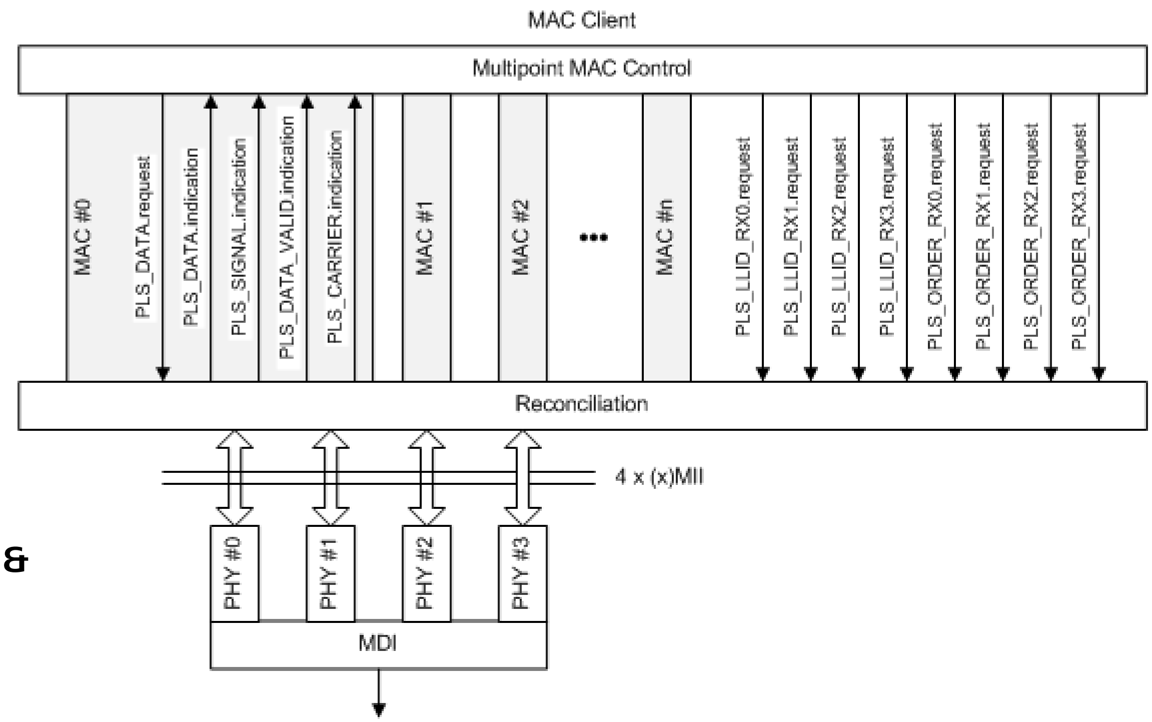


See 802.3 subclause 21.5 for state diagrams conventions



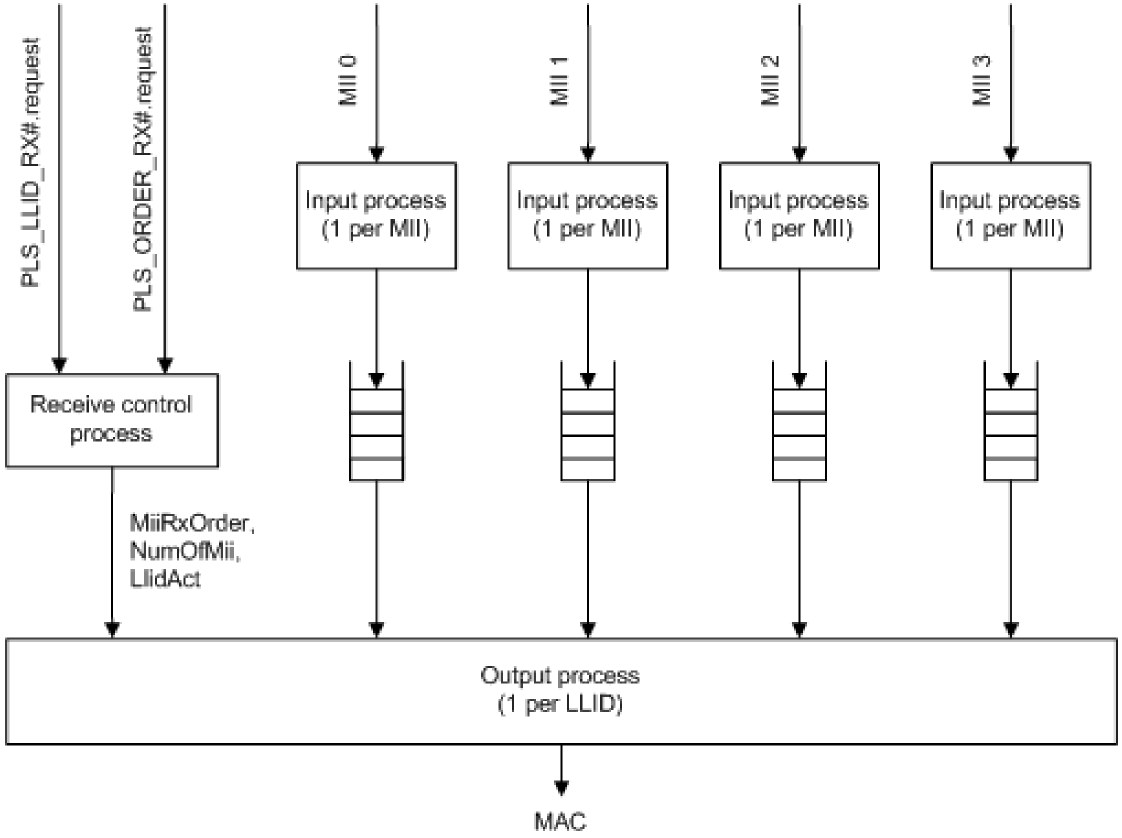
OLT architecture

- **In addition to typical PLS signals include signal to communicate the granting pattern**
  - PLS_LLID_RX#
    - **LLID info**
  - PLS_ORDER_RX#
    - **Chanel and channel order**
    - **Ex. For reception on ch 2, 1, & 3 (in that order)**
      - **PLS_ORDER_RX0 = 0 of 0 (no data)**
      - **PLS_ORDER_RX1 = 2 of 3**
      - **PLS_ORDER_RX2 = 1 of 3**
      - **PLS_ORDER_RX3 = 3 of 3**



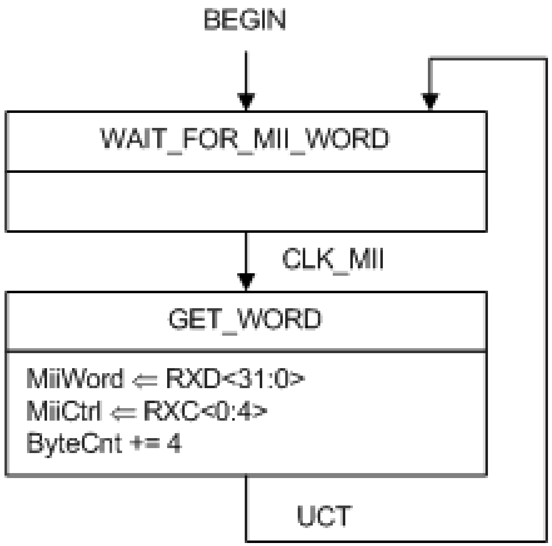
OLT block diagram

- **Three processes**
  - Input process – once per Channel / 25GMII
  - Receive control process
  - Output process
- **One buffer (fifo) per channel**



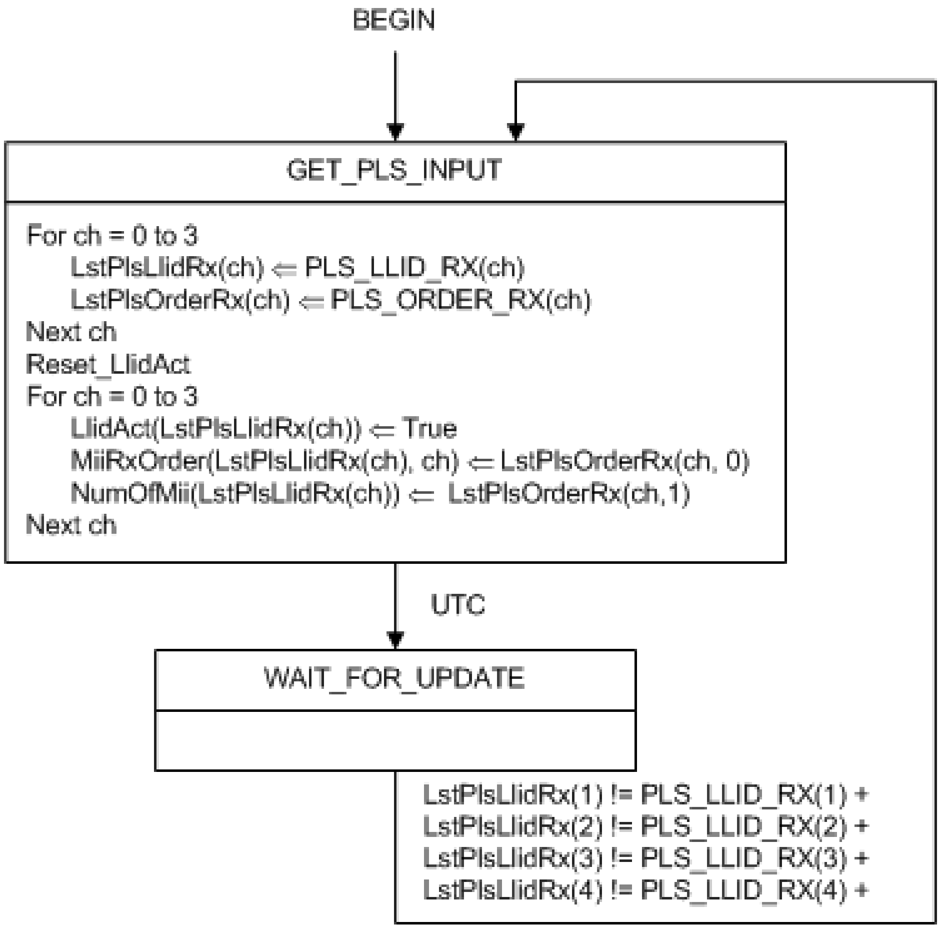
Input process (1 per 25GMII)

- **CLK_MII – clear on read Boolean that goes True on each edge of the MII clock**
- **MiiWord – 32-bit binary array that temporarily storages an Mii data word**
- **MiiCtrl –4-bit binary array that temporarily storages an Mii control nibble**
- **ByteCnt – number of bytes stored in the channel buffer**



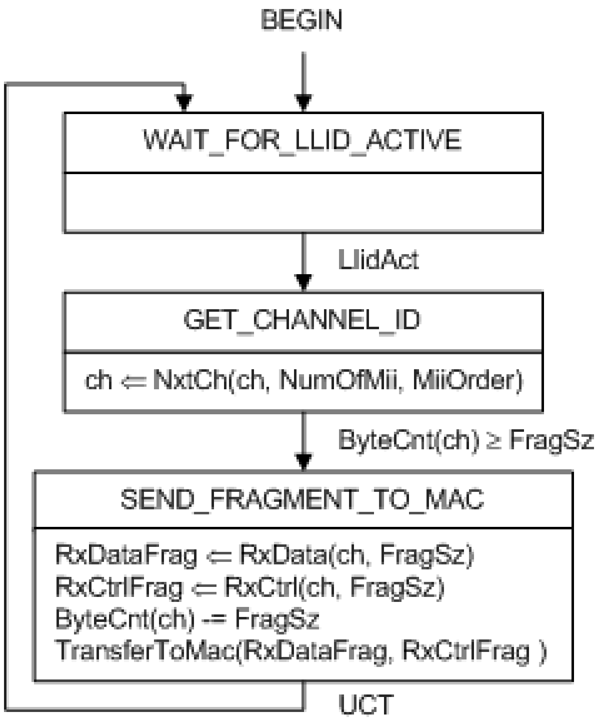
Control process

- LlidAct(l) – When True this Boolean indicates that LLID (l) is actively receiving frames.
- LstPlsLlidRx(n) – this integer holds the last recorded value of PLS_LLID_RXn where n is the channel number.
- LstPlsOrderRx(n,m) – this ordered set holds that last recorded value of PLS_ORDER_RXn where n is the channel number. If m = 0 the channel order number is returned if m = 1 the number of channels is returned, if m is omitted both channel and number of channels values are returned.
- MiiRxOrder(l, n) = this integer holds the channel order number (n) for the llid identified by l.
- NumOfMii(l) – this integer holds the number of MII channels in use for the LLID identified by l.
- Reset_LlidAct – this function sets LlidAct(l) to False for every valid value of l.



Output process (1 per LLID)

- ch – an integer representing the channel
- FragSz – a constant that is equal to the size of a fragment.
- NxtCh(c, n, m) – this function returns the next channel in the receive order specified given the current channel number (c), the number of channels in use (n) and the order of channels (m).
- RxData(c, s) – a byte array of data read from buffer (c) of size (s) bytes.
- RxDataFrag – this byte array is a temporary holder for the next data fragment to be sent to the MAC.
- RxCtrl(c, s) - a bit array of control information read from buffer (c) of size (s) bits.
- RxCtrlFrag – this bit array is a temporary holder for the control bits associated with each byte in RxDataFrag.
- TransferToMac(d, c) – this function shifts out a byte array (d) to the MAC consistent with the definition of the 25GMII or XGMII as defined in Clause 6 & 46 and control information (c). The function does not return control to the calling state until all bits have been shifted to the MAC.



Conclusions

- Channel bonding and fragmentation identification mechanism in kramer_3ca_1_0916 has lots of advantages and support.
  No reason to change that.

- **But** … the OLT could reinforce that mechanism with the ideas presented herein to provide early detection and recovery from potential errrors in the data stream.
- Good material for an informative annex.

Thank you

www.huawei.com

What about skew?

- **If we pick our channel wavelength right we can use wavelength induced skew to our advantage**
- **Ex (calculated at 40 km)**
  - Delay Ch1 ≥ Ch2 ≥ Ch3 ≥ Ch4
  - This would be reinforced by a "load high order channels first" rule

Custom Grid wavelengths				
1	2	3	4	Channel
1283.00	1288.00	1293.00	1298.00	Min
1287.00	1292.00	1297.00	1302.00	Max

Maximum optical skew	14.4	UI
Maximum optical skew variation	6.1	UI