



**XILINX**

ALL PROGRAMMABLE™

# FlexEthernet - Protocols and Components

**Mark Gustlin**  
**Principal System Architect**  
**October 16<sup>th</sup> 2014**

# Introduction

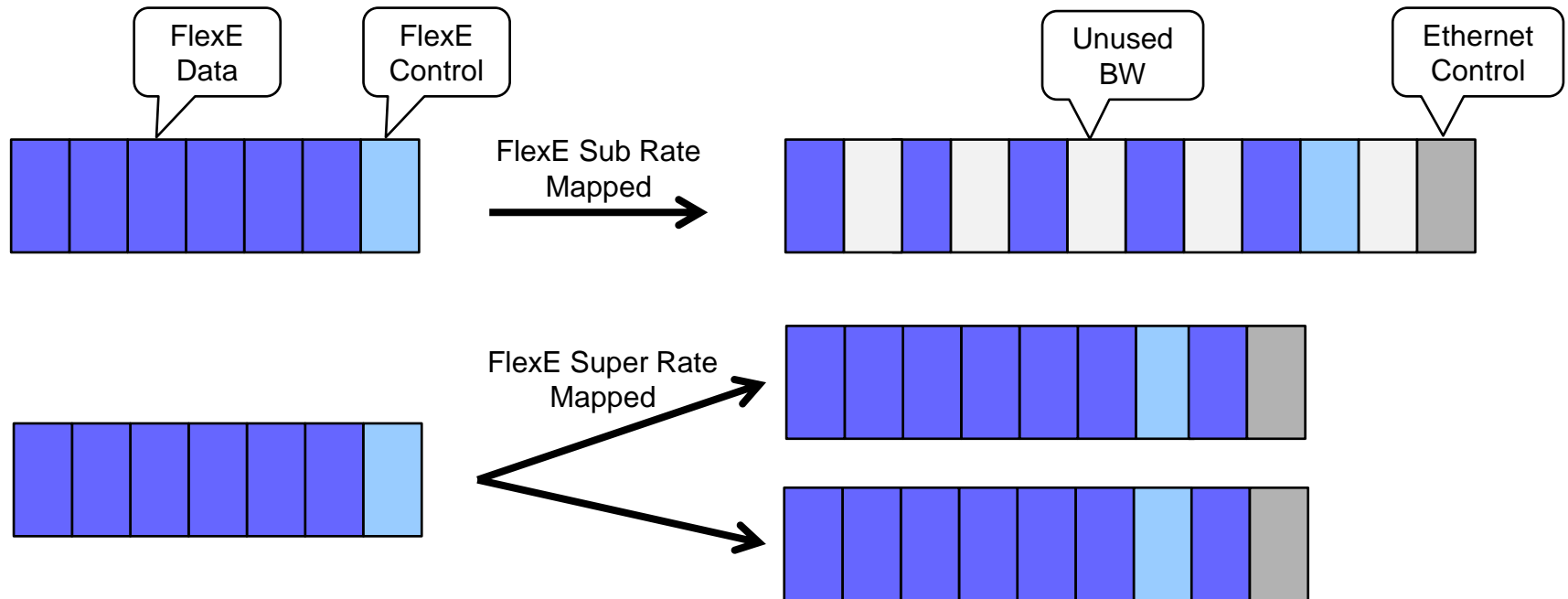
- **The Goals of the FlexEthernet protocol definition are:**
  - Support the required use cases
  - Simple and efficient protocol that can run over existing and future Ethernet PMDs and electrical interfaces without change to the PMDs or interface definitions
- **The next few slides explore the current thinking about the protocol and how that impacts IEEE PMDs/Interfaces**

# Thoughts on Current Protocols

- Initial thinking on FlexEthernet was to re-use/or make it similar to MLG
- MLG is an example of good opportunistic re-use of an existing industry protocol
  - MLG 1.0 and 2.0 re-uses the 100GbE Physical Coding Sublayer (PCS)
  - Supports carrying nx10GbE and nx40GbE over nx25GbE lanes
- Then 802.3bj came along and added RS-FEC to 100GbE
  - The FEC is required for many copper and optical interfaces now, backplane, multi-mode fiber and some MSA optics
  - Using RS-FEC requires transcoding and some manipulations of the Alignment Markers
  - The PHYs as defined in 802.3bj and 802.3bm cannot transparently carry MLG 1.0 or 2.0, MLG is in place of the PCS, does not run over a standard PCS
  - This required a new MLG version (3.0) which is currently being developed
- Would be better if future MLG can be carried transparently over any current or future Ethernet PMD/PHY
- And translated to FlexE, we should design FlexE to be carried transparently over future new PMDs/Interfaces

# FlexEthernet's Flexibility

- Goal of the protocol, in addition to supporting the desired use cases, is to run transparently over any current or future IEEE PMD or electrical interfaces (PHYs)
- Each time a new PMD or interface is defined, we don't want to be compelled re-define FlexE
- What does this mean?
  - To Ethernet FlexE should look like data or control which is then transparently carried
  - No reliance on FlexE knowing what IEEE is doing, nor should any IEEE PMD/Interface care about what FlexE is doing



# FlexEthernet's Flexibility

## ➤ What complications are there?

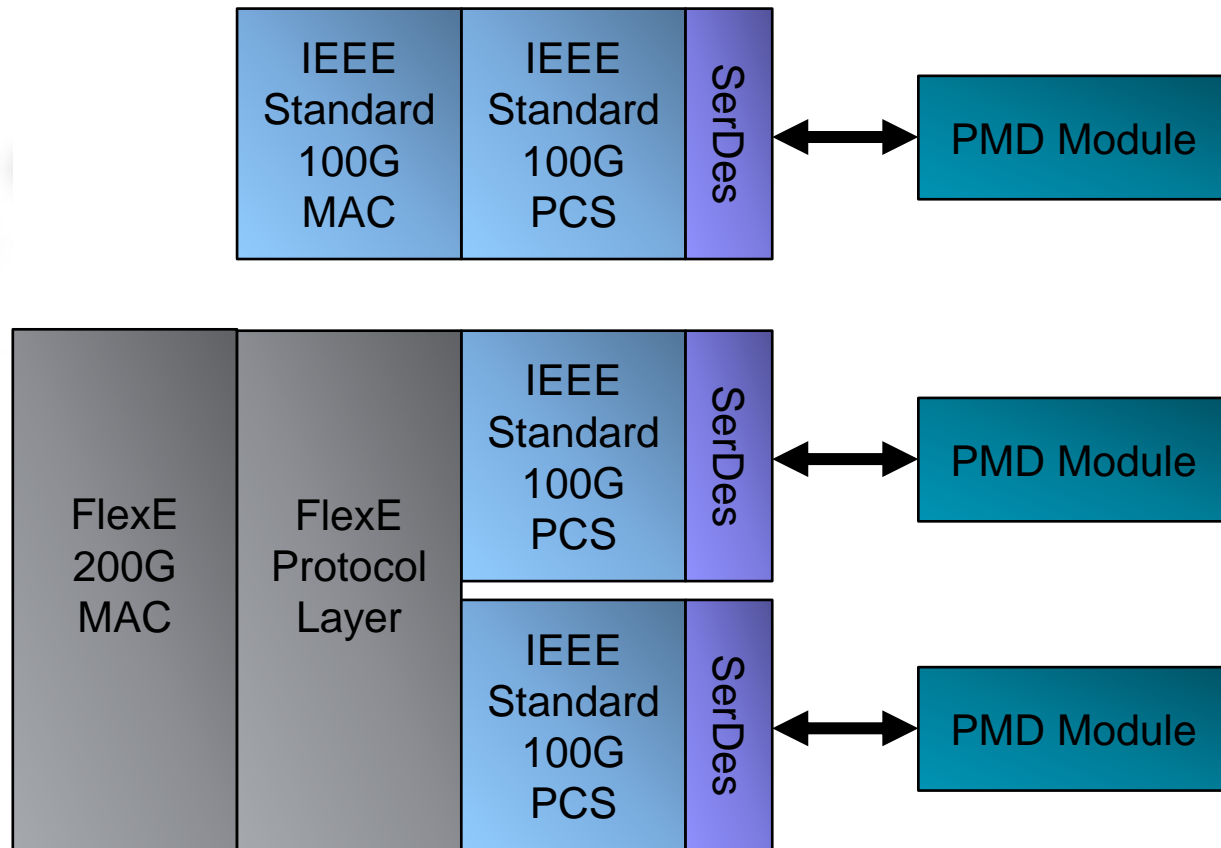
- We still want a simple protocol, without complicated floating payloads with pointer processing for instance
- Deleting/inserting idles is what is used in 100GbE interfaces for adding in alignment mechanisms
- If we have to add/remove alignment or other mechanisms at multiple layers of an implementation, how do we do that without dependencies on those layers?
- This is especially a problem if the complete data path is carried over multiple physical paths, you don't always have the full view of the packet data to delete Ethernet idles

## ➤ How to solve these concerns?

- Looking at defining a unique FlexEthernet framing structure to delineate FlexEthernet data
- Exploring unique FlexEthernet Idles (in place of Ethernet Idles) to allow flexible rate adaptation along a FlexEthernet path
  - A given FlexEthernet stream might not have enough information to be able to delete Ethernet Idles

# MAC/PCS Impacts

- You can re-use standard IEEE PCS implementations, and add on the FlexEthernet protocol and the FlexEthernet MAC



# PMD Impacts

## ➤ The goal is to use IEEE PMDs as is for FlexEthernet

- For example a 100GBASE-LR4 or 100GBASE-SR4 optics module with accompanying PHY protocol stack can be re-used without modification
- MSA defined PMDs should also work without modification, for example CWDM4
- So no impact!

## ➤ What this implies to the FlexEthernet protocol:

- Same per lane rate on the PMDs
- Therefore we must delete extra idles from the MAC stream to make room for the FlexEthernet overhead in addition to the normal overhead used for a given IEEE PHY
  - Normal IEEE overhead is Alignment Markers for multi-lane interfaces
- FlexEthernet must run transparently through the IEEE PHYs, so must look like data so it survives transcoding unaltered



**Thanks!**