ESIstream

The **E**fficient **S**erial **I**nterface

Implementation Note 01: Deterministic Latency

Version 1.0

ESIstream is an Open high efficiency serial interface protocol based on 14b/16b encoding.

Its main benefits are low overhead and ease of hardware implementation.

TABLE OF CONTENT

FIGURE TABLE

DETERMINISTIC LATENCY OPERATION

The ESIstream protocol is compatible with deterministic latency requirements. There are 2 different cases to consider.

1.1 System with 1 lane

In this case the latency of the transmission is composed of the TX latency, the lane delay and the RX latency.

The TX latency depends on the implementation of the TX but is fixed.
The lane delay depends on the external physical channel between TX and RX.
The RX latency also depends on the RX implementation but is fixed.

Thus for a system with a single serial lane, the latency of the transmission is deterministic.

1.2 System with multiple lanes

In case of multiples lanes, the latency is composed of the same parts: TX latency, lane delays and RX latency. However, the lane delays and the RX latency may differ between lanes.
The lane delay will depend on the PCB trace length, the skew between the lanes. One of the advantages of serial interfaces is that the constraint on the trace lengths is much smaller compared to parallel interface. In fact, it is only limited by the RX buffer size and the trace attenuation. The possibility of having different trace lengths on a system impacts the lane delay and additional processing needs to be implemented so that after synchronization, all lanes are aligned at the output of the RX stage.

To achieve that with ESIstream, the receiver has to integrate a buffer. This buffer will add a variable latency which needs to be determined for systems that require latency information.

One solution would be to use the synchronization to align all lanes together and determine the latency on each lane.

Reminder: Synchronization sequence:

Figure 1: Synchronization sequence

The idea is that on each lane, the transfer from the frame alignment synchronization sequence to the PRBS initialization sequence would be recovered. When it is found on a lane, its data starts to be written in the buffer. When it has been found on all lanes, the data starts to be read from the buffer. This ensures that all lanes are aligned at the output of the buffer. The buffer can also be used to separate clock domains.

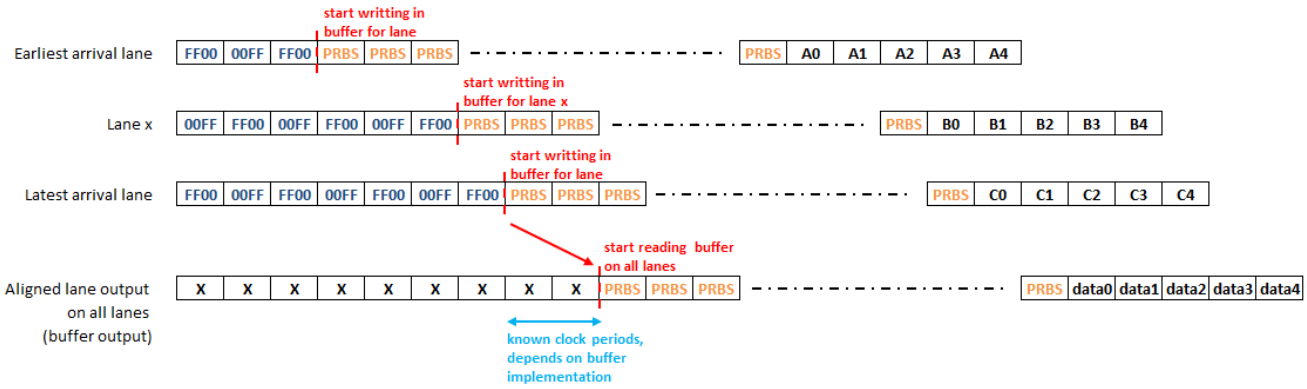

Figure 2: Timing diagram illustration for deterministic latency

At the output of the buffer, after synchronization, all lanes are aligned. They have a known and fixed latency depending on the lane trace lengths difference.