EICROC Interface Document
Between ASIC Designers and FTOF/FF Users

Tonko Ljubicic, Rice Uni
27 Mar 2024
Early DRAFT

Introduction

General Comment: It should be noted that the ASIC will directly interface to a Xilinx Ultrascale+ family of FPGAs.

We assume the ASIC only requires 1.2V.

We want to avoid buffers or any other need for signal translation schemes to enable a simple design with lower power, smaller area and less cost. The Xilinx nomenclature calls the single ended standard LVCMOS12 while the differential standard is DIFF_SSTL12_DIV.

We hope to use the smallest possible signal count on the ASIC to be able to make a robust design where up to 64 ASICs are interfaced to a single Readout Board.

Input Clock to the ASIC (the “ASIC Clock”)

We expect that the **only** clock required by the ASIC is a multiple of the 98.5 MHz EIC clock but generally as slow as possible to make the overall system design simpler and more robust especially with longer cables.

It will be transmitted to the ASIC over 1 differential lane according to the XXX signaling standard. NB: the clock will not originate from the FPGA but will be driven directly by a fanout PLL e.g. Si53302.

The signal termination will be at the ASIC end. The lane is expected to be AC coupled on the PCB.

This clock will be synchronized in frequency and also in phase to the EIC collisions. We expect the random jitter of this clock to be less than 5 ps.

Output Data from the ASIC (“ASIC Data”)

Signaling

We expect 1 or more differential data lanes carrying the serial data from the ASIC to the FPGA. The ASIC should be able to readout the data of all channels even on 1 serial link so the number of links in use could be made programmable but default to 1 at power-up.

The serial data rate should be a multiple of the ASIC Clock with the slowest data rate being equal to the EIC 98.5 MHz clock. If the data rate is programmable it should default to the slowest data rate possible.

The ASIC signaling standard should be capable of directly interfacing to the Xilinx DIFF_SSTL12_DIV signal standard. The termination will be on the FPGA end.

It would be beneficial to the overall readout design if the data links can be daisy-chained ASIC-to-ASIC (over e.g. 2-4 ASICs) which would then simplify designs where we plan to readout up to 64 ASICs with the same RDO.

Expected Data Format per Hit

10 bits ADC
10 bits TDC, perhaps 11?
10 bits BCID (Bunch crossing)? Perhaps 16?
10 bits channel ID [0..1023]
xx bits other?

1) Encoding, DC Balancing? Raw or 8b10b or … ? AC or DC coupled?
2) Will there be any checksum or checkum-like data?
3) What is the IDLE token on the serial links?
4) Any header/trailer in case of multiple hits per Bunch Crossing?
5) BCID counter counts using which clock?
6) Since we are in Streaming Mode, what defines the time of the start (or value=0) of the TDC? Which edge of which clock?

Expected Data Rate

**For the following calculations I will assume 64 bits per hit which is in line with other typical ASICs.*

30 Hz noise, 3 Hz signal ⇒ 33 Hz hits per channel.
1024 channels ⇒ 34 kHz hits per ASIC

64 bits/hit ⇒ **2.2 Mb/s per ASIC** (note how low this is)

⇒ Alex comment: What about other EICROC detectors???
- Roman Pots
- B0
- Luminosity Detector? Likely not EICROC though…

Fast Commands to the ASIC

Fast Commands are transmitted to the ASIC over a single dedicated differential pair. The termination will be at the ASIC end.

Some important commands will be issued by an EPIC-wide entity synchronously to the EIC collisions & clock at the same time (same EIC crossing) for all detectors so the response of the ASIC needs to be deterministic.

1) We require that the commands be synchronous to the ASIC Clock or its multiple.

What is the expected clock multiple, or rate?
Number of bits/command?
Encoding?
DC-balancing? AC/DC coupled?

2) Commands

a) IDLE (when there is no command)
b) CLEAR_COUNTER (clear BCID counters)
c) Others?

3) It would greatly simplify the production readout scheme if the “Fast Command” link can be either daisy-chained from ASIC to ASIC (e.g. to 2-4 ASICs) or multi dropped (e.g. to 2-4 ASICs)

I2C Control/Status

Number of bits for the I2C address encoding is 7 for 64 possible devices in a single chain. Can be made smaller for FTOF (5 bits).

Naturally, one can’t realistically expect 128 devices on a long signal cable so what is a reasonable number of I2C address bits for the ASIC designers?

We assume the highest feasible clock rate (SCL) is 1 MHz.

It would be efficient if the “Fast Command” data lane can be used instead of I2C due to its much higher speed. At least in production systems. I2C can be kept for debugging or during single ASIC testing.

Other

We require that the ASIC doesn’t *need* ***any other signals*** for production operation.
Adding other required signals (e.g. various resets, etc) complicates production designs where we have up to 64 ASICs connected to 1 Readout Board.