

ALCOR v3 - digital interface

- Clock frequency: **394.1 MHz** (4x EIC clock: 98.52525 MHz)
- **8 LVDS Tx links**: one for each column, DDR output at **788 Mb/s**
- Data format
 - **32-bit words** with **8b/10b encoding** (2x event words in ToT/SR mode)
 - Frames with *header + data + trailer*
 - Frame duration set by *coarse counter roll-over* or *New Orbit reset*
- External **reset** with 3 different features depending on signal width:
 - 24-31 xCLK: **Hard Reset**
 - 16-23 xCLK: **Start** (Frame Counter = 0 and Coarse Counter = 0)
 - 8-15 xCLK: **New Orbit** (Frame Counter +1 and Coarse Counter = 0) → forces a *rollover* condition (ALCOR rollover = 2^{15} clk cycles $\approx$ 83 μ s, EIC orbit period $\approx$ 12.8 μ s)
- **Test Pulse**: analogue FE injection and TDC calibration; also used for **shutter** signal; propagated from EoC to each pixel
- **SPI** interface for registers configuration

FIFO position	Data (32-bit word)
1	K28.0 (Frame header)
2	Frame number
3 + ...	Event words Column X
n	K28.2 (Frame trailer)
n+1	Frame info (frame_len, ev_lost)
n+2	<i>K28.3 (Status header)</i>
n+3	<i>Status words Column X (8x)</i>
n+11	<i>End of Column status word</i>
n+12	K28.4 (Checksum header)
n+13	CRC value

Single channel data format example (event words only)

Frame	Index	Data	Pixel	TDC id	TCoarse	TFine	Timestamp (coarse ns)
28	0	0008feaf	0	0	047f	0af	2296637.5000
28	0	010900ca	0	1	0480	0ca	2296640.0000
30	0	0212fe8a	0	2	097f	08a	2463677.5000
30	0	031300ac	0	3	0980	0ac	2463680.0000
31	1	00019681	0	0	00cb	081	2540027.5000
31	1	0101989d	0	1	00cc	09d	2540030.0000
31	1	02229e8c	0	2	114f	08c	2550597.5000
31	1	0322a0ae	0	3	1150	0ae	2550600.0000
34	0	0103e661	0	1	01f3	061	2786527.5000
34	0	0003e6c5	0	0	01f3	0c5	2786527.5000
34	0	031bb661	0	3	0ddb	061	2794147.5000
34	0	021bb6be	0	2	0ddb	0be	2794147.5000
35	1	0003ce8e	0	0	01e7	08e	2868417.5000
35	1	0103d0a9	0	1	01e8	0a9	2868420.0000
35	1	03198665	0	3	0cc3	065	2875367.5000
35	1	021986c2	0	2	0cc3	0c2	2875367.5000
35	1	0020d691	0	0	106b	091	2877707.5000

} $\Delta = 100 \text{ clk}$

} $\Delta = 93 \text{ clk}$

- Data from trailing edge (TDC 1, TDC 3) can be transmitted before data from leading edge (TDC 0, TDC 2)
- Data from next bunch can be transmitted before data from previous bunch

ALCOR data generation and transmission

Pixel TDC conversion defines time required for digitization:

- TDC max conversion time = $1.5 \times 128 = 192$ clock cycles ≈ 500 ns
- TDC min conversion time = $0.5 \times 128 = 64$ clock cycles ≈ 170 ns
- Max $\Delta T = 500$ ns - 170 ns = 330 ns $\rightarrow$ this is inside the ALCOR channel, then you need to propagate the data along the column and transmit off-chip

ALCOR data transmission:

- 1 word: 40 bits $\rightarrow 40 \text{ b} / 788 \text{ Mb/s} = 51$ ns per event word (we have 8 channels for each Tx link)
- If we have no other hits in the column, 2 hits (separated by 128 clk cycles in the pixel) are only separated by idle words (0xBCBCBCBC), so **this strongly depends on data rates**
- Idle words and other filtered by RDO?
- Time reference given by frame structure: $\text{DCR} = 300 \text{ kHz/ch}$, $\Delta t_{\text{frame}} = 12.7886 \mu\text{s} \rightarrow N_{\text{events}} = 8 \cdot \text{DCR} \cdot \Delta t_{\text{frame}} = 30.7$
- 31 event words per frame (mean value, 2x in ToT mode, add physics) $\rightarrow$ can they be ordered in FPGA by Tcoarse bits?

Data format:

- $\Delta t_{\text{frame}} = 12.7886 \mu\text{s} \rightarrow 13$ bits from Tcoarse are enough ($2^{13} \cdot 2.5 \text{ ns} = 20.4 \mu\text{s}$) $\rightarrow$ we can remove 2 bits

Summary

- **ALCOR data-push architecture** implies that events are not transmitted perfectly ordered by time
 - Need to consider **TDC conversion time** (128 clk max delay) but also **data transmission** (50 ns per event word)
 - Time reference is provided by **frames structure** ($\Delta t_{\text{frame}} = 12.7886 \mu\text{s}$) and typically (offline) you need to consider a full frame to merge and reconstruct data
- Do we want to merge 2 event words (in FPGA) when we are in ToT/SR mode?
- We can provide **data stream example files**, but we need input files from physics simulations to have more realistic data output (DCR + physics + other)