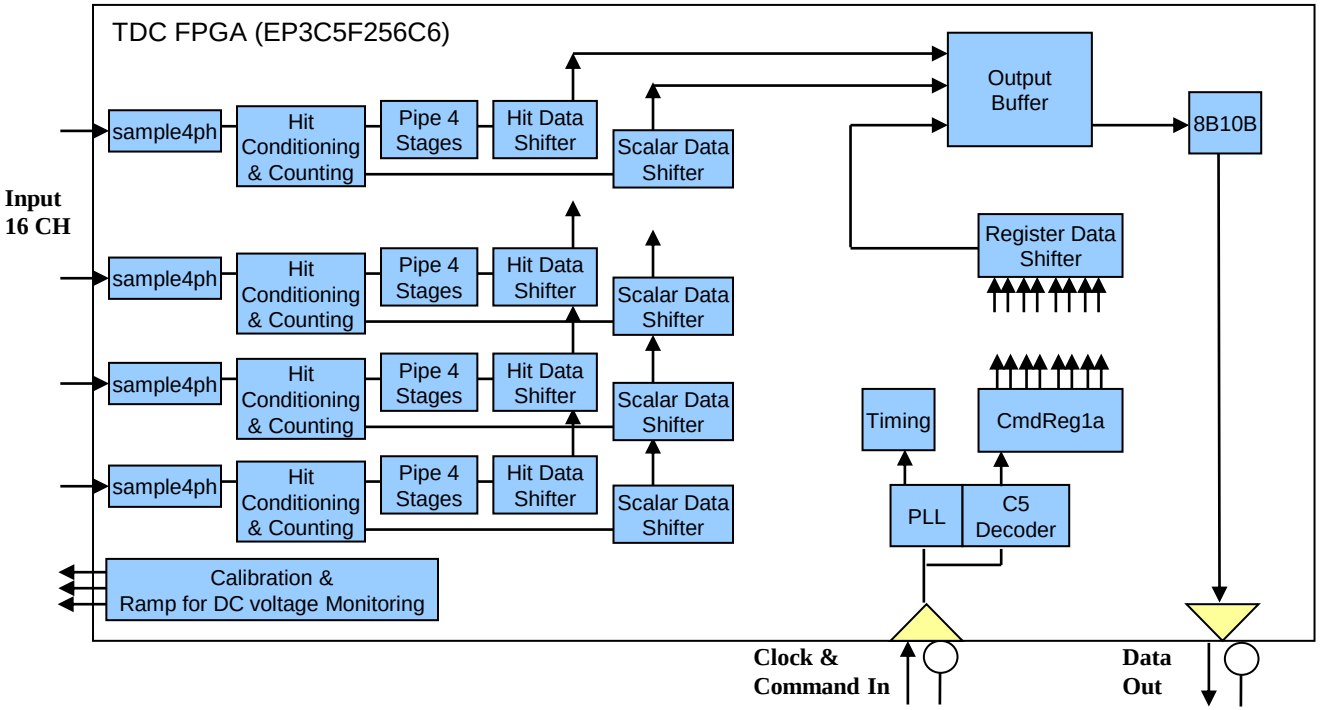
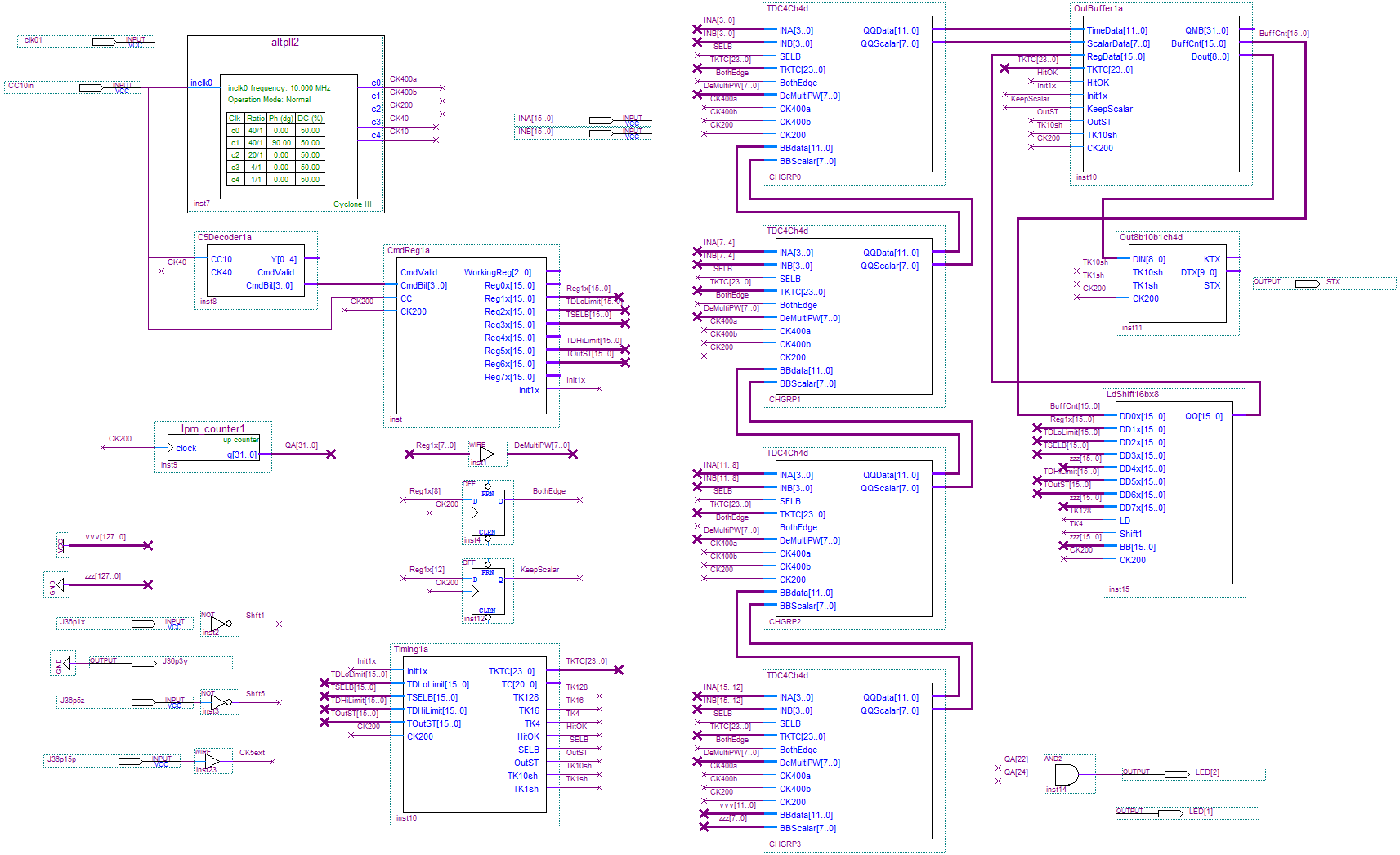
Block Diagram

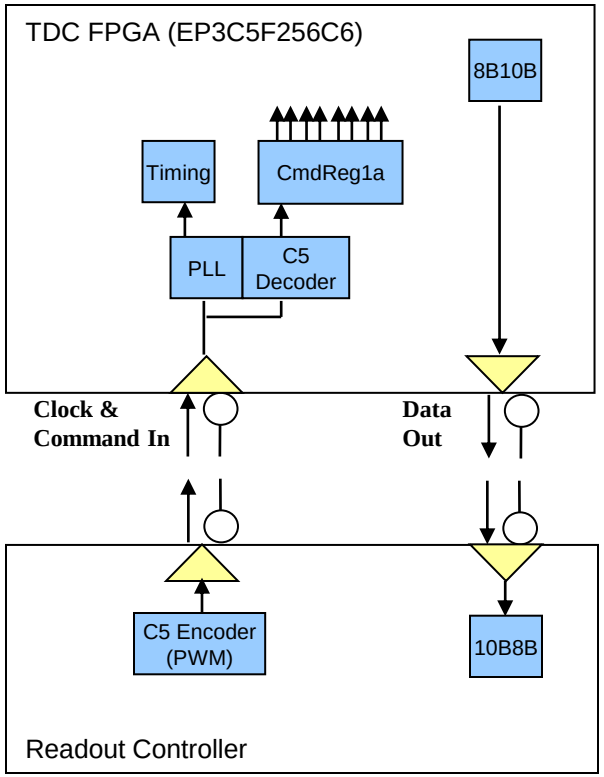


- A 16-channel TDC fits in EP3C5F256C6 with ~50% silicon resource usage.
- Input clock at 10 MHz is multiplied to 200 MHz and two phases of 400 MHz for internal operation.
- Both positive and negative input transitions are digitized with precision LSB 625 ps. (=1/(4*400 MHz))
- Each channel is equipped with a 32-bit scalar which keeps hit count across long period of time.
- For each 10 ms spill, up to 2016 TDC hit data, scalar data, internal register data and an ID header are packed into 2048 32-bit words and are output using 8B10B protocol at 25 Mb/s.
- Bursts of up to 4 very rapid hits can be captured. Rapid multiple hits can also be eliminated in the hit conditioning block based on users' setting.
- Only two LVDS pairs, Clock & Command In and Data Out are needed from readout controller to serve the FPGA.
- Calibration hits generation and DC voltage monitoring features are provided.

Actual Top Page

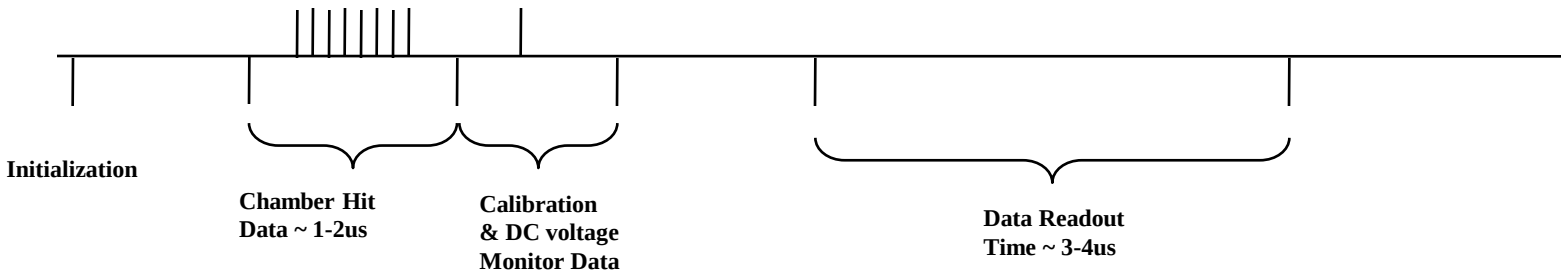


Interconnections



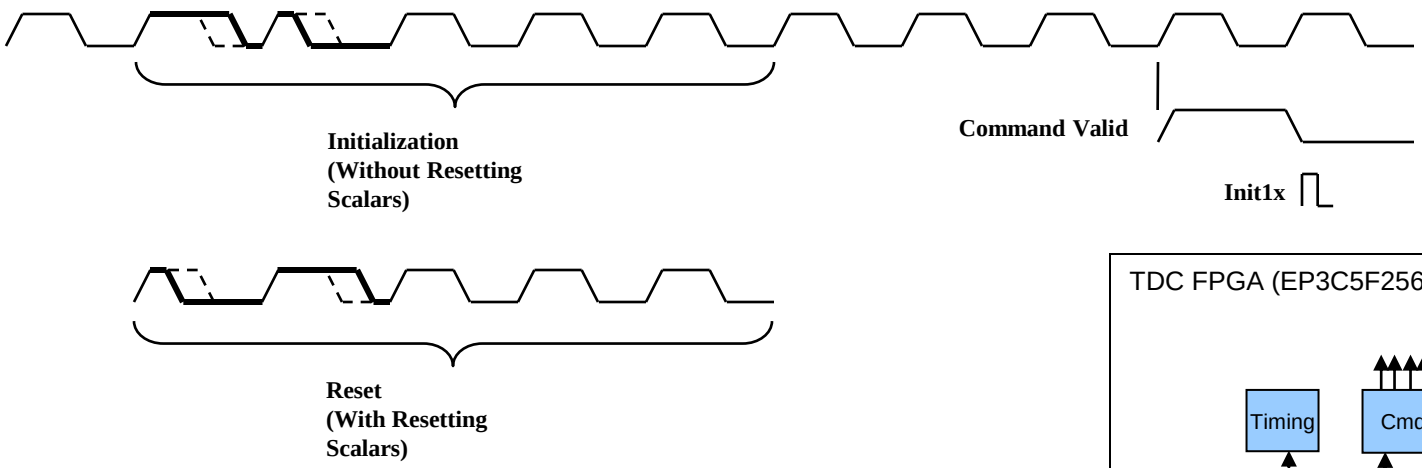
- Only two pairs of fast signals are needed between TDC and the readout controller. (Extra wires in the cable can be used for JTAG or other FPGA reconfiguration signals.)
- The Readout Controller sends 10 MHz clock pulses to drive the TDC.
- Register setting and initialization commands are sent via pulse width modulation via the C5 Encoder and C5 Decoder.
- Data from TDC FPGA is a 25 Mb/s 8B/10B stream.
- It is decoded in the Readout Controller.
- The encoder and decoders are based on over sampling scheme. No dedicated transceivers are needed.

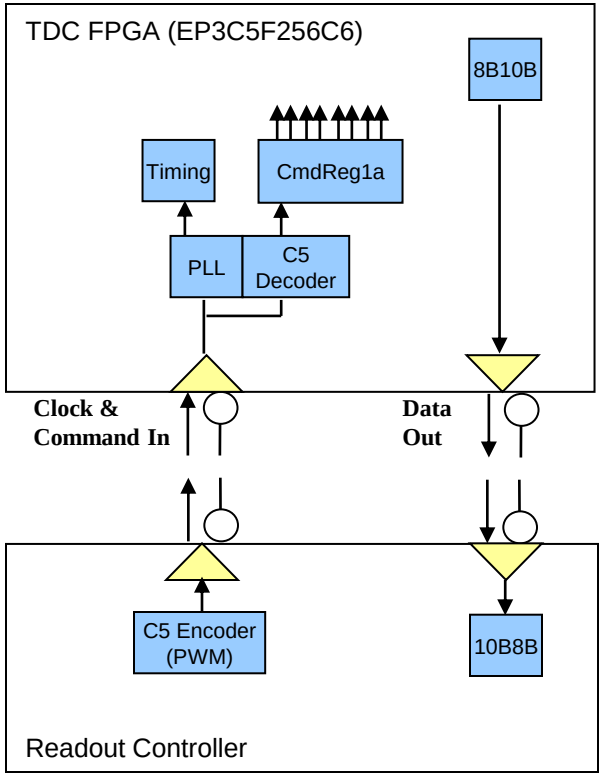
Normal Operation



- At the start of the 10 ms spill period, a initialization command is sent to the TDC FPGA that reset the operating sequence.
- Chamber hit data are collected during the user specified time range.
- After collecting chamber hits, TDC values for calibration and DC voltage monitoring are collected.
- Approximately halfway of the spill period, data are sent out to the readout controller.
- The TDC FPGA then stays idle state until being initialized again for next spill.

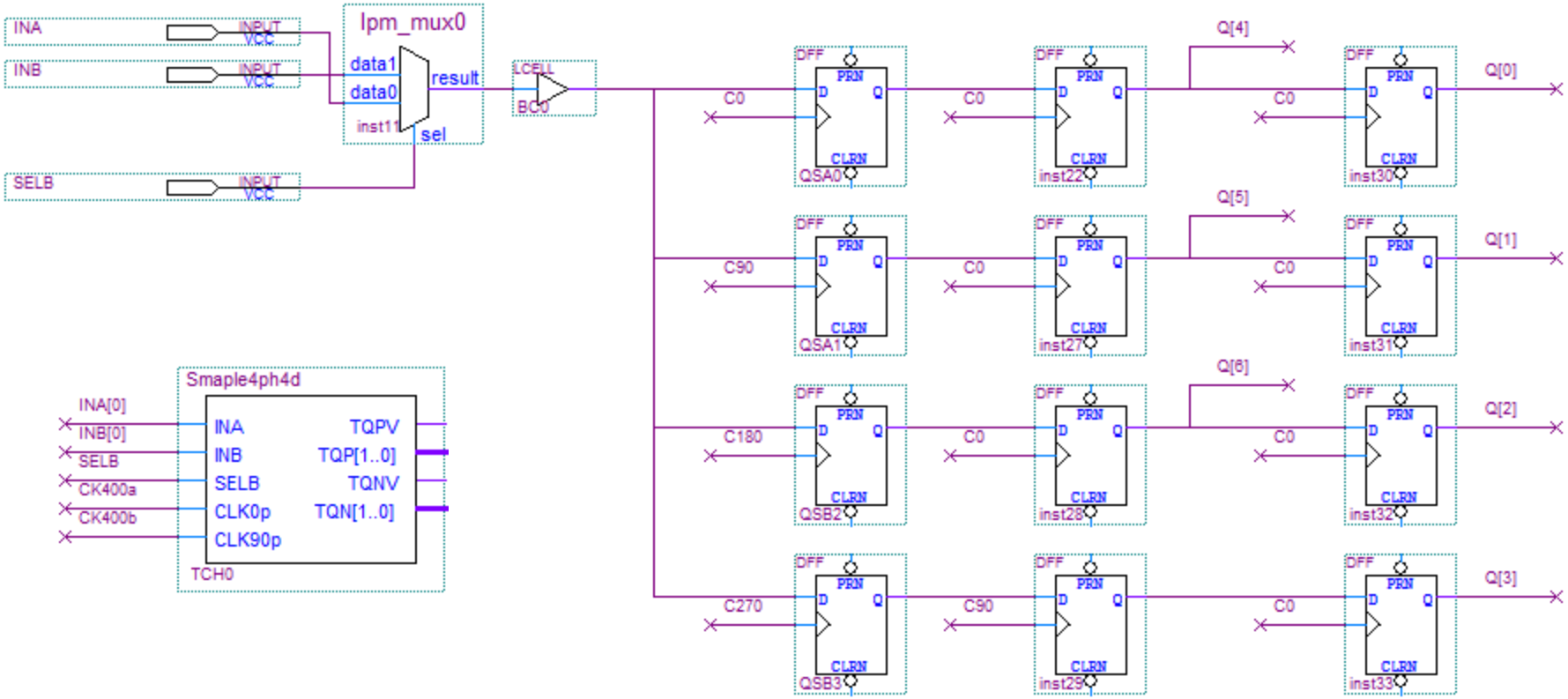
Initialization of Spill





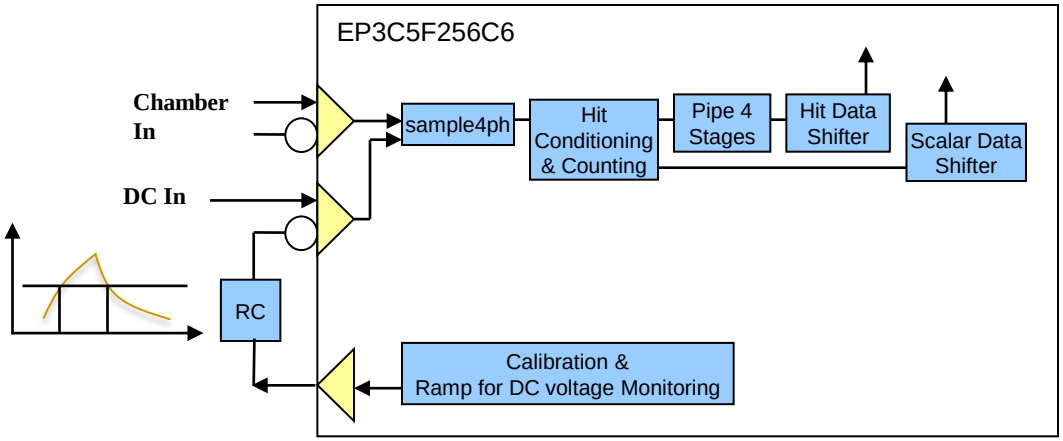
- A wide-narrow sequence is decoded as a initialization command without resetting scalars.
- After a known latency, an initialization pulse is generated inside FPGA that resets the coarse time counter and a normal operation sequence is started.
- The narrow-wide sequence can be reserved as resetting command that will reset scalars.

The 4 Phase Sampler



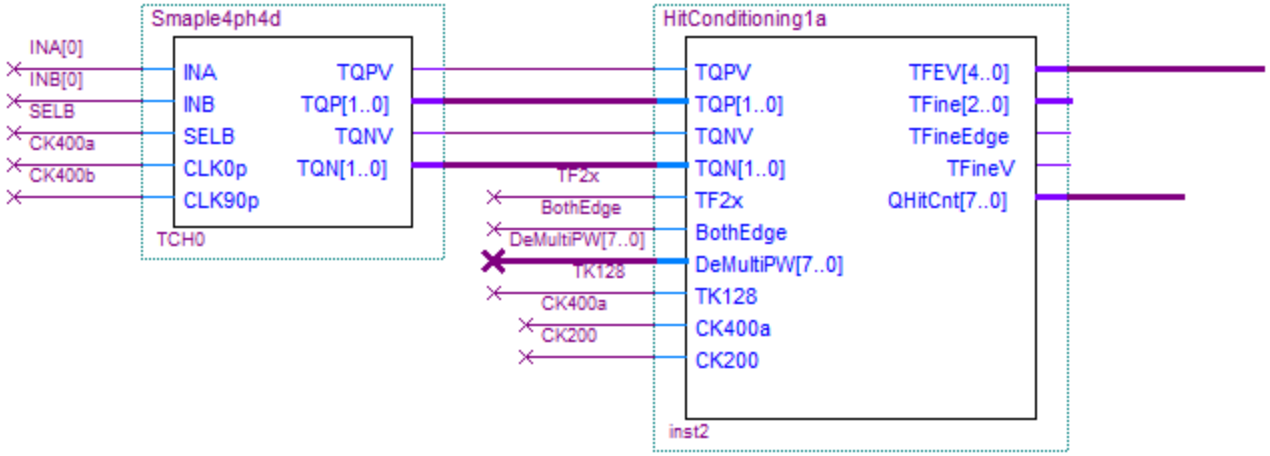
- The inputs are sampled with 4 phases of 400 MHz clocks.
- The samples are transferred to the c0 clock domain and are encoded.
- The TDC fine time LSB is 625 ps.
- Both positive and negative transition edges are encoded but the users are allowed to choose output mode positive edge only or both edges in the hit conditioning stage.
- Two inputs are fed into the TDC: INA from the chamber and INB for DC voltage monitoring.

DC Voltage Monitoring



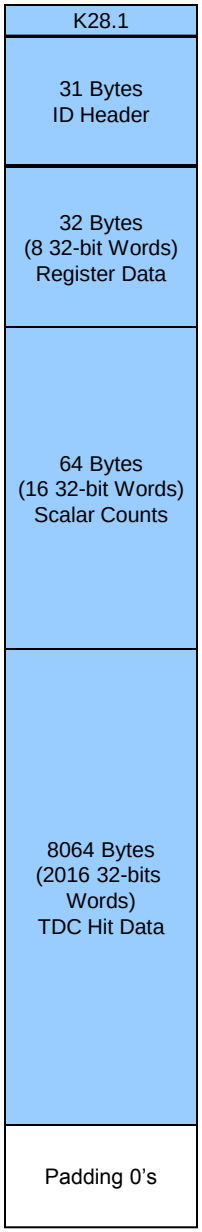
- The + end of the LVDS pair of the B input is connected to the DC voltage (0.5-2.0V) to be monitored.
- A 2.5 V logic level is generated by the FPGA.
- The RC network turn the logic level into a reference ramping voltage.
- The crossing time of the ramping voltage and the DC voltage is digitized by the TDC block and merged into data stream.

Hit Conditioning



- The hit conditioning block performs several preprocessing functions on the hits from the sampling block.
- The clock domain is transferred from 400 MHz to 200 MHz to reduce design difficulties in the later stage.
- A counter is implemented to count number of hits (scalar support).
- Depending on users' preference, the data from negative transition are either kept or eliminated:
  - BothEgde = 1, keep both transitions; =0, keep only positive transitions.
- The users are allowed to generate a hit inhibit signal to eliminate additional hits for a user specified period of time after receiving a hit:
  - DeMultiPW[6]=1, Enable multi-hit elimination; =0, Disable multi-hit elimination.
  - DeMultiPW[5..0]= 0-31, number of 5 ns cycles to inhibit further hits.
  - DeMultiPW[6]=1, Updating mode; =0, non-updating mode.

Output Data

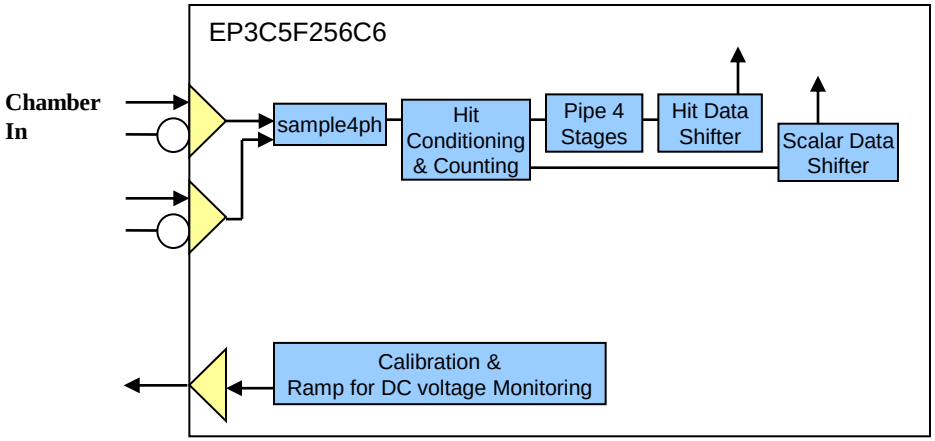


- Data are sent out in 8B/10B serial format at 25 Mb/s.
- Each output sequence starts with a K28.1 code so that the receiver can be aligned.
- The ID Header contains some constants set in the firmware.
- The Register Data, Scalar Counts and TDC Hit Data are organized as 32-bit words.
- After sending out all data, the transmitter keeps sending out 0's to maintain correct DC balance.

Data Bit Layout

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
K code & ID block 0-7	0																								0x3C, K28.1							
	0																								0x67 = ‘g’							
	0																								0x2D = ‘-’							
	0																								0x32 = ‘2’							
	0																								0x54 = ‘T’							
	0																								0x44 = ‘D’							
	0																								0x43 = ‘C’							
	Version date, e.g., 0x20130401																															
Internal Register 8-15	0																Register 7															
	0																Register 6															
	0																Register 5															
	0																Register 4															
	0																Register 3															
	0																Register 2															
	0																Register 1															
	0																Register 0 = Number of non-empty 32-bit words															
Scalar Readout 16-31	Scalar Count 15																															
	Scalar Count 14																															
	Scalar Count 0																															
Hits 32-2047	0	0	1	Edge	CH: 0-15				Coarse Time: LSB= 5 ns, Range = 0 - 10.485775 ms																					Fine Time LSB=625ps		
	0	0	1	Edge	CH: 0-15				Coarse Time: LSB= 5 ns, Range = 0 - 10.485775 ms																					Fine Time LSB=625ps		

Calibration Scheme



- ?

Registers

Register	Bit(s)	Function
Reg0x		Not used
Reg1x	7..0	DeMultiPW; Pulse width for de-multi-hit function, 0-255 x 5 ns
	8	BothEdge
	12	KeepScaler
Reg2x	15..0	TDLoLimit[15..0]
Reg3x	15..0	TSELB[15..0]
Reg4x		Not used
Reg5x	15..0	TDHiLimit[15..0]
Reg6x	15..0	TOutST[15..0]
Reg7x		Not used

Registers Setting

CmdBit[3..0]	
2	Shift in 0
3	Shift in 1
4, 12	Initialize micro-bunch
9, 10, 11, 13, 14	Select Working register 1, 2, 3, 5, 6

- To set register, send C5 codes (D0-D15) into the input.
- Select working register first by sending D9, D10, D11, D13, D14.
- D2 feeds a bit with value 0 into the LSB of the register.
- D3 feeds a bit with value 1 into the LSB of the register.
- D4 and D12 are micro-bunch initialization commands.