IBL PPC scan structure

The loop structure of the IBL ROD scan engine follows essentially the strategy from the DSP scan engine of the current Pixel ROD [1]. Note that the following differs from the original proposal [2] by having parameter loop and double-column (DC) loop swapped.

The meaning of the processing loops ordered from innermost (i.e. executed first) to outermost (i.e. executed last) is as follows – see figure 1:

- **Trigger loop**: sending the requested number of CAL (strobe) and LV1 (trigger) signals, waiting for the EFB (taking into account feedback from the histogrammer to the EFB by RFD-signal) to have finished after each CAL-LV1-pair which needs PPC-EFB-interaction.
- **DC loop**: turning on the charge injection mechanism on a sub-set of the DCs in order to not overload the charge pump. Typically, scans using analogue injection would turn on one DC at a time. Digital scans will not need this loop at all but instead turn all DCs on at the same time (see section 8.3 in Ref. [3]). NB: this is not related to any pixel register action even though the same global registers are used for DC selection.
- **Parameter loop**: alters the actually scanned variable which is typically a register on the FE (e.g. the amount of injected charge, PlsrDAC) or a data taking parameter (e.g. the trigger delay). Data is transferred to DDR after each parameter step, while data shipping from DDR to fit server can be done after several parameter steps or even the entire loop in order to allow fitting in parallel to data taking. Due to initial memory limitations, data of at most 1/8 of all connected FE can be stored before transfer to DDR in case all FEs are connected to a slave such that only 8-step-masks (see below) must be used for now. Possible mask patterns need to be revised once external memory is available.
- **Mask loop**: use the enable pixel register, optionally with the injection capacitor pixel registers for analogue-injection-based scans or the shift register for digital-injection-based scans (see table 26 in Ref. [3]), to turn on a sub-set of all pixels in a regular pattern. An n-step-mask means that every n-th pixel is turned on, starting at rows 1,n+1,2n+1,... in even columns, and starting at rows n,2n,3n,... in odd columns[1]. At each mask step, the masks are shifted to lower row indices in odd, to higher row indices in even columns[2], see figure 2 and command sequence below. The mask pattern is communicated to the fit server at the start of the scan.

Note that all global and pixel register operations in the scan loop can and should normally be sent in broadcast mode to all FEs in parallel, i.e. the MSB of the 4-bit chip-ID field is high (see sec. 6.3.2 in Ref. [3]). In addition, pixel register writing and shifting of the regular mask pattern should be done to all DCs at the same time (Colpr_Mode is set to 3 (binary 11), Colpr_Addr is set to 0). The pixel register is written only for mask step 0; all subsequent mask steps use the shifting mechanism:

1. Set threshold high (global register 20, Vthin_Fine and Vthin_Coarse). Not needed for pure shifting functionality, but safety measure to keep disturbance of the FE low.
2. Activate all DCs for pixel register operation: in global register 22, Colpr_Mode is set to 3, Colpr_Addr is set to 0 (due to overlap of the Colpr-register with the charge pump activation, this must be done *each* time).
3. Select pixel register to be shifted via pixel latch strobe in global register 13 (one by one, i.e. the entire procedure has to be repeated if several registers need to be shifted).

[1] Rows and columns in FE-I4 are counted starting from 1, see figure 2.

[2] Pixel register writing and shifting is done per DC such that odd and even column of a DC are connected at row 1, see figure 2.

4. Latch pixel register into shift register: set S0=S1=1 (global register 13), HitLd to 0 (global register 21), LatchEn to 1 (global register 27); send global pulse.
5. Latch content of shift register back into pixel register: set S0=S1=0 (global register 13), HitLd to 0 (global register 21), LatchEn to 1 (global register 27); send global pulse.
6. Repeat step 4; this combined with step 5 applies inversion of the content of the pixel register during read back twice to get the original (non-inverted) content.
7. Move shift register content by one pixel: set S0=S1=1 (global register 13), HitLd to 0 (global register 21), LatchEn to 0 (global register 27) and send *one* global pulse *per pixel* to be shifted.
8. Latch content of shift register back into pixel register (step 5).
9. Set threshold back to original value (global register 20, Vthin_Fine and Vthin_Coarse).

A typical example is a threshold scan which requires a set of occupancy histograms, one per value of injected charge, to which an S-Curve-fit is applied. A scan with 100 events per charge value, charge being scanned from PlsrDAC=0,...,200 in steps of 1, a single-DC-loop and an 8-step-mask is processed as follows:

1. General configuration, histogrammer set-up etc.
2. Enable and both capacitor pixel registers are initialised enabling rows 8,16,34,... in columns 1,3,5,... and enabling rows 1,9,17,... in columns 2,4,6,... (same regular pattern written to all DCs at the same time). The histogrammer is informed about mask step 0.
3. PlsrDAC is set to its first value of 0 by writing to the according global register part.
4. 1st DC is enabled: Colpr_Mode is set to 0, Colpr_Addr is set to 0 (same global register).
5. 100 strobe (CAL pulse) and trigger commands are sent to the FE while data are being histogrammed.
6. 2nd to 40th DC is enabled (Colpr_Addr = 1,...,39), repeating item 5 each time.
7. Data from histogrammer memory is shipped to the fit server and slave histograms are cleared.
8. PlsrDAC is incremented by one (PlsrDAC=1,...,200), repeating items 4 to 7 each time.
9. The content of the enable and both cap. pixel registers is shifted by one, i.e. enabling rows 7,15,33,... in columns 1,3,5,.. and rows 2,10,18,... in columns 2,4,6,... The histogrammer is informed about mask step 1 and items 3 to 8 are processed again. Repeat this item 7 times such that each pixel is enabled once.
10. Execute general scan termination commands. Ship fit results from the fit server to the histogram server (PPC or PixLib action?).

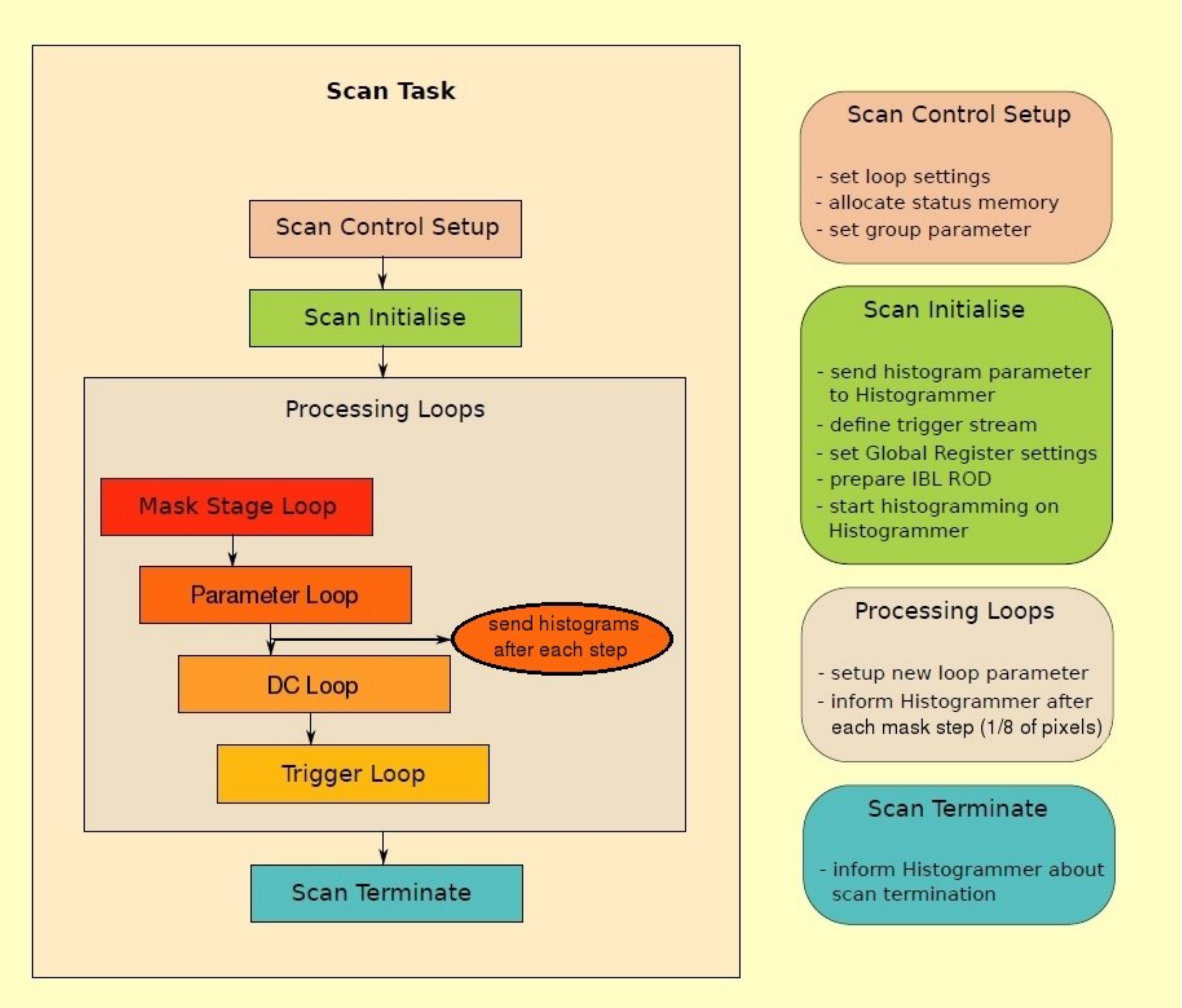

Figure 1: Structure of the scan procedure on the IBL ROD PPC. *Modified from [2], sec. 7.2.2*

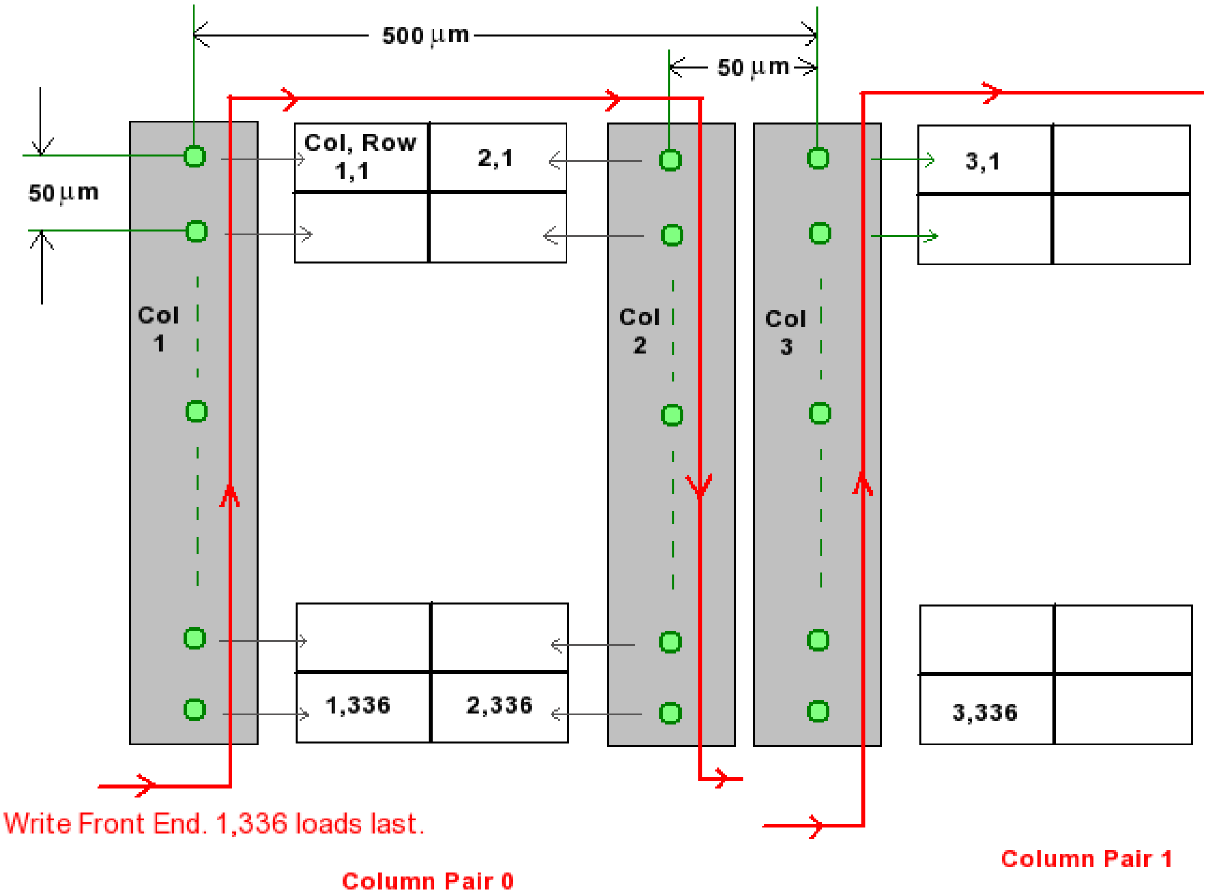

Figure 2: Pixel and column numbering scheme. The red line shows the pixel configuration shift register flow. Each double-column has an independent shift register. To program the pixels in columns 1 and 2 one must select double-column 0 for shift register programming, etc [3].

References

[1] J. Biesiada et al., *The Implementation and Performance of ROD DSP Software in the ATLAS Pixel Detector*, ATL-INDET-INT-2009-006.

[2] N. Krieger, *Development of the readout for the IBL Upgrade Project of the ATLAS Pixel Detector*, PhD thesis Uni Göttingen II.Physik-UniGö-Diss-2012/08.

[3] FE-I4 Collaboration, *The FE-I4B Integrated Circuit Guide*, version 2.3 on IBL share point.