Missing hits investigation

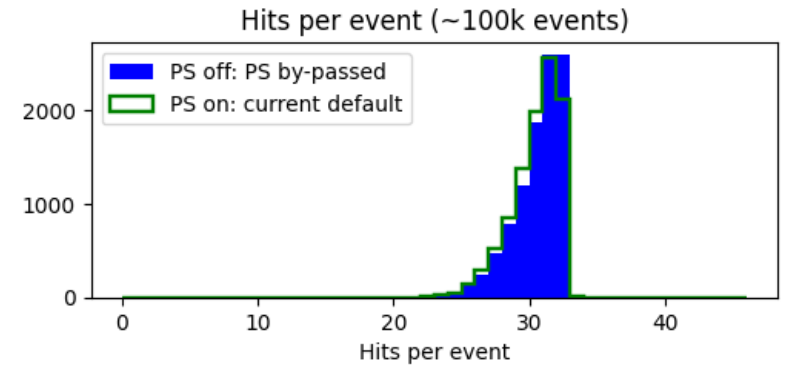
- Position loss due to corrupt data is still under investigation. This is probably not the main source of the missing hits.
- The special case of hits that span two windows had a problem causing loss of position. This has been fixed in v3.3 candidate.
- Another bug was the check for data corruption related to windowspan cases. Will be fixed in 3.4.
- Fixing both issues, restores the missing hits.

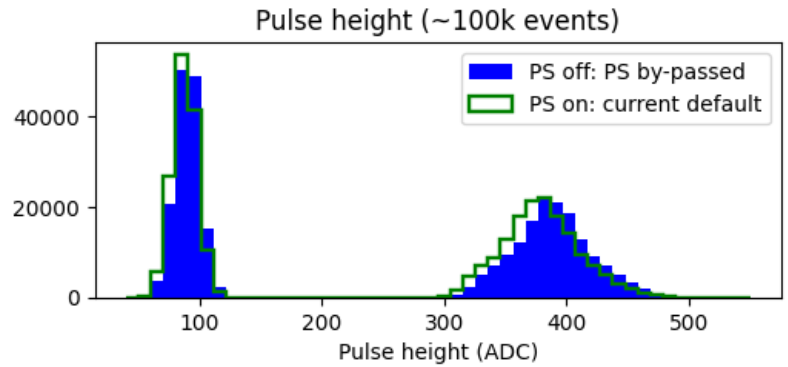
```
276 +
277 +            for (samp = startSample; samp < startSample+16 && samp < 64; samp += 2) {
278 +              int16_t loAdc = swapEndian(ch_det_buf_in[pos]) & 0xFFF;
279 +              int16_t hiAdc = (swapEndian(ch_det_buf_in[pos]) >> 16) & 0xFFF;
280 +              if (debug) {sprintf(errmsg, "%d, %d ", loAdc,hiAdc); printfMessage(errmsg);}
281 +              if (pedestal_subtraction_flag != 0) { //Will put back after enabling pedestal subtraction
282 +                loAdc -= pedestal_val[bs_index+baseIndex+samp];
283 +                hiAdc -= pedestal_val[bs_index+baseIndex+samp+1];
284 +                if(pedestal_val[bs_index+baseIndex+samp]<600 || pedestal_val[bs_index+baseIndex+samp+1]<600 ||
 pedestal_val[bs_index+baseIndex+samp]>1500 || pedestal_val[bs_index+baseIndex+samp+1]>1500){ //using the same condition as firmware
285 +                  hitInfo.integral = 1;
286 +                }
287 +              }
288 +              // int16_t *samplePtr = (int16_t *) &(ch_det_buf_in[pos]);
289 +              // samplePtr[0] = loAdc;
290 +              // samplePtr[1] = hiAdc;
291 +              sampleData[samp-startSample] = loAdc;
292 +              sampleData[samp+1-startSample] = hiAdc;
293 +              if(writeRawHit){FullWaveformsToCopy[CounterFullWaveformsToCopy][2+(samp-startSample)/2] = swapEndian((loAdc << 16) | hiAdc);}
294 +              pos++;
295 +            }
296 +
297 +
```

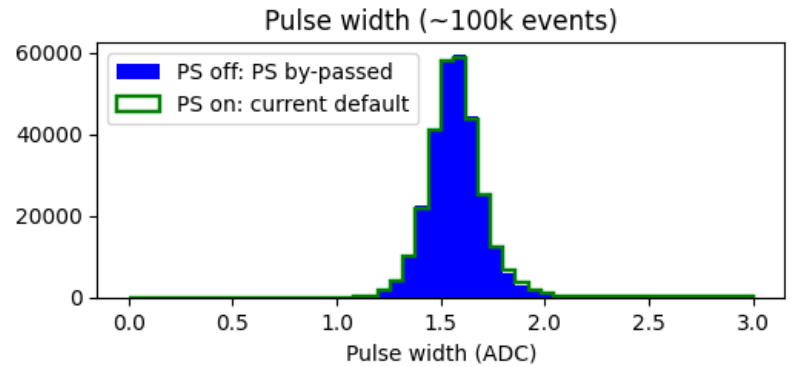
```
if (nSamples != 32 && nSamples != 16) {
  nSamplesWrongCounter++;
  if (debug)
  {
    sprintf(errmsg,"ASIC %ld hit %ld incorrec
    sprintf(errmsg,"\thad totHits = %ld, char
  }

  totHits -= nHits - iHit;

  while (swapEndian(ch_det_buf_in[pos]) != 0x
    pos++;
  }
  if (debug) {sprintf(errmsg,"stopped at worc
  ProcessPacketProblemCounter++;
  //pos++;
  break;      You, 5 months ago • Initial
  //continue;
}
```

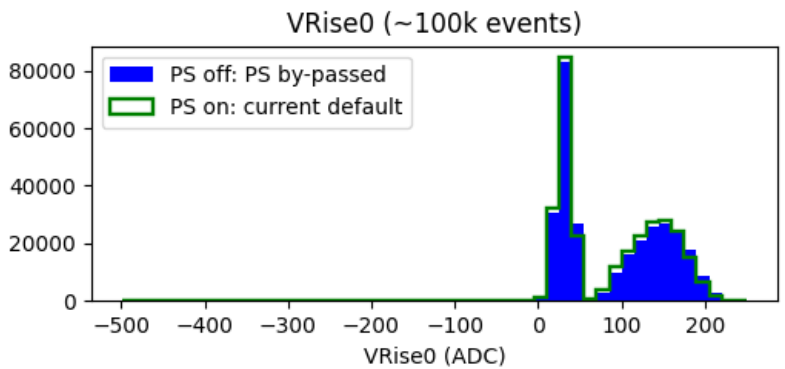
PCIe40 FE Dead time

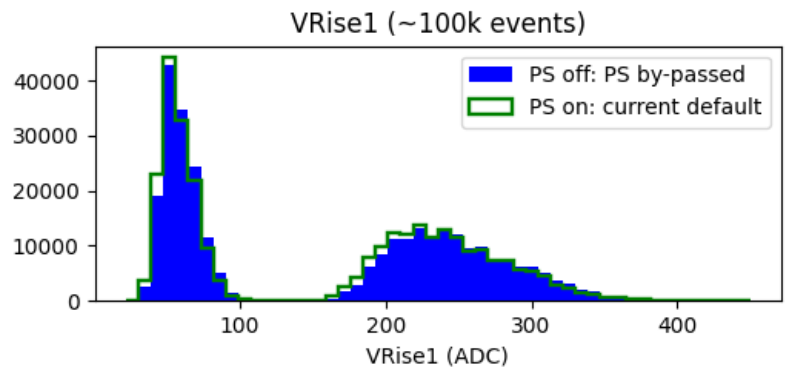
- The dead time observed by Harsh is under investigation. Dead time is not distributed near uniformly between TOP boardstacks. We have three candidates main for explanation in terms of likelihood:
  - One or more TOP boardstacks due to bad analogue threshold or some other reason generate too much noise and saturate the connection.
  - Individual channel problems on the PCIe40 board or computer.
  - The shared memory space for the TOP data on PCIe40 is filled and through an unknown mechanism an asymmetric backpressure creates dead time on individual TOP boardstacks.

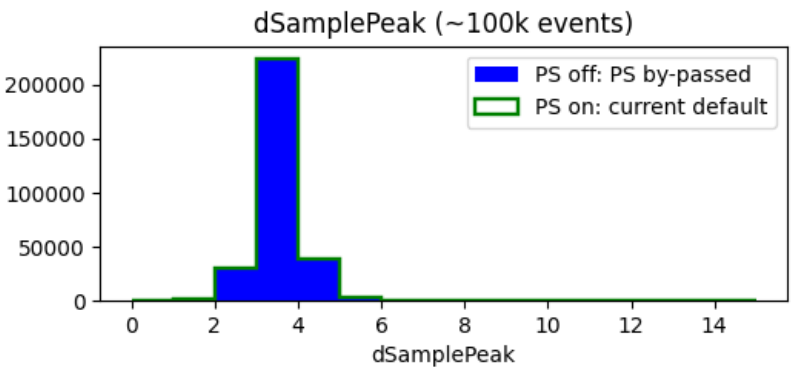

Hits per event (~100k events)
PS off: PS by-passed
PS on: current default
Hits per event

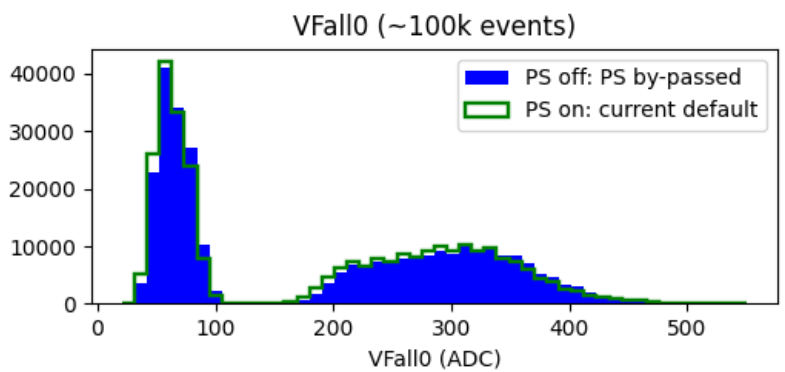

Pulse height (~100k events)
PS off: PS by-passed
PS on: current default
Pulse height (ADC)

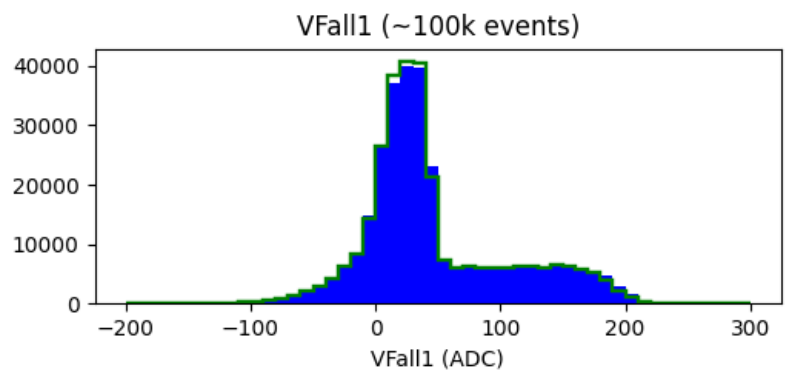

Pulse width (~100k events)
PS off: PS by-passed
PS on: current default
Pulse width (ADC)

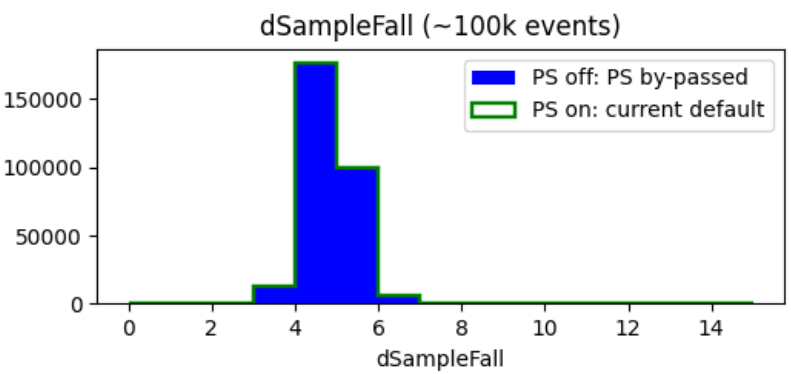

VRise0 (~100k events)
PS off: PS by-passed
PS on: current default
VRise0 (ADC)


VRise1 (~100k events)
PS off: PS by-passed
PS on: current default
VRise1 (ADC)


dSamplePeak (~100k events)
PS off: PS by-passed
PS on: current default
dSamplePeak


VFall0 (~100k events)
PS off: PS by-passed
PS on: current default
VFall0 (ADC)


VFall1 (~100k events)
PS off: PS by-passed
PS on: current default
VFall1 (ADC)


dSampleFall (~100k events)
PS off: PS by-passed
PS on: current default
dSampleFall