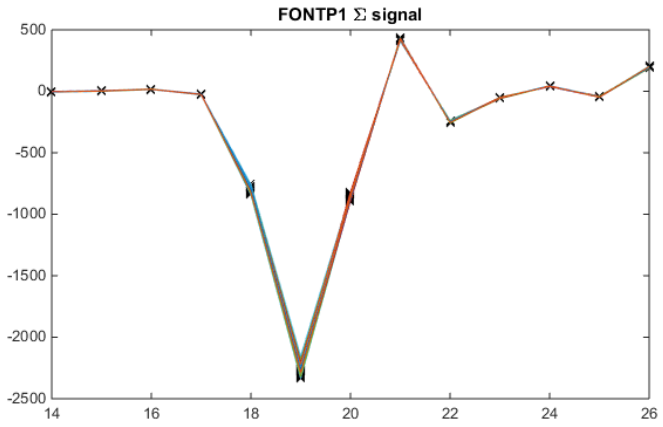
Stripline BPM resolution + Firmware Plans

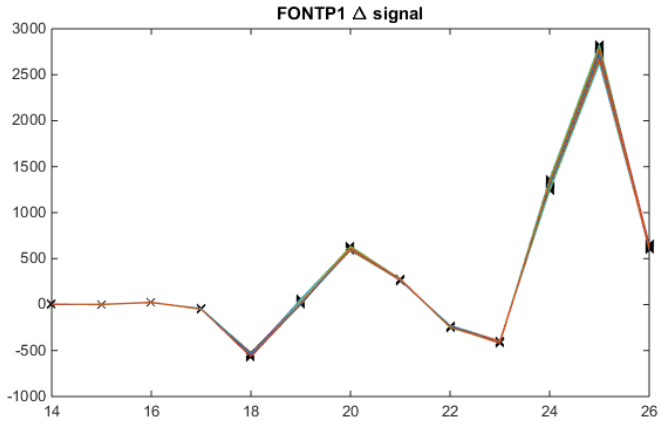
Glenn Christian

26/08/16

Effect of inclusion of off-peak samples

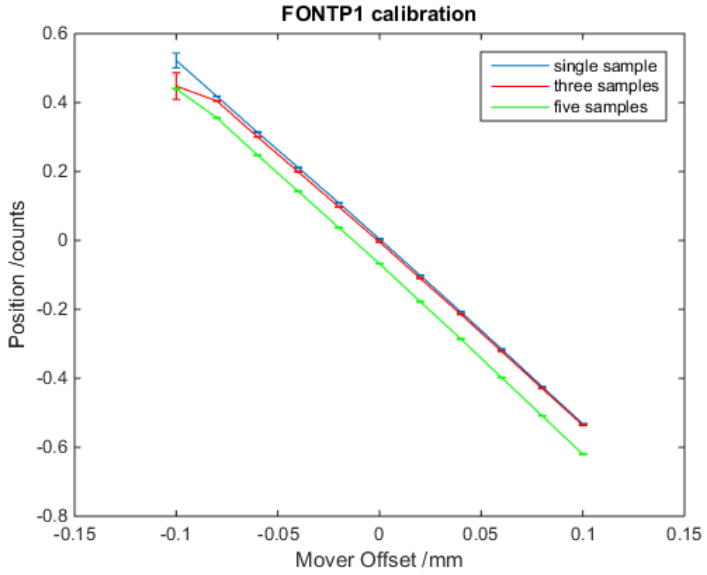
- Looked at some time ago – negligible effect, but resolution generally poorer ~1 micron
- Naively, expect at best ~3 dB improvement in resolution

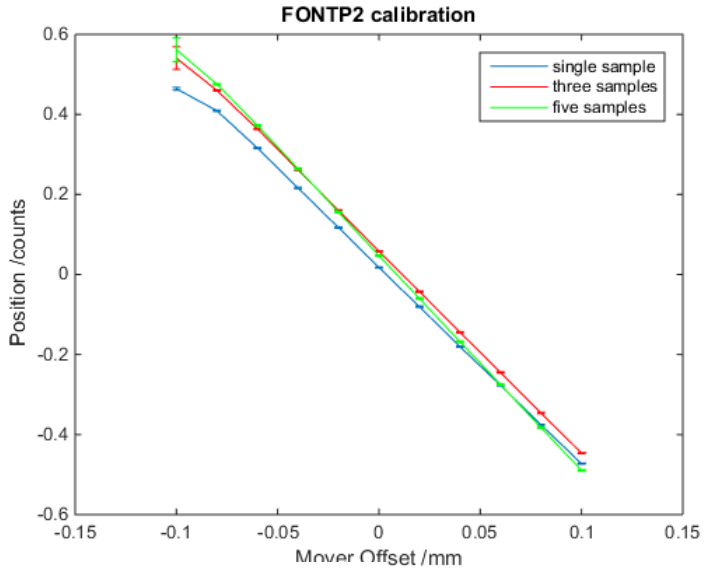


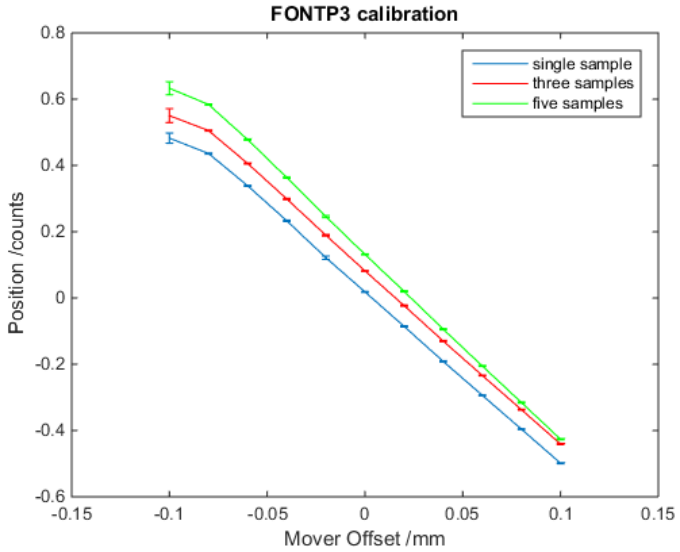


Data: mCal5_MoverVPos_0.000_Board1_100616

Calibration check
(mCal3 on 100616)







Calibration constants with Integration

Integration width	FONTP1	FONTP2	FONTP3
1	-0.0052	-0.0049	-0.0052
3	-0.0052	-0.0051	-0.0053
5	-0.0054	-0.0054	-0.0057
7	-0.0053	-0.0054	-0.0055
9	-0.0054	-0.0055	-0.0056
11	-0.0053	-0.0053	-0.0058

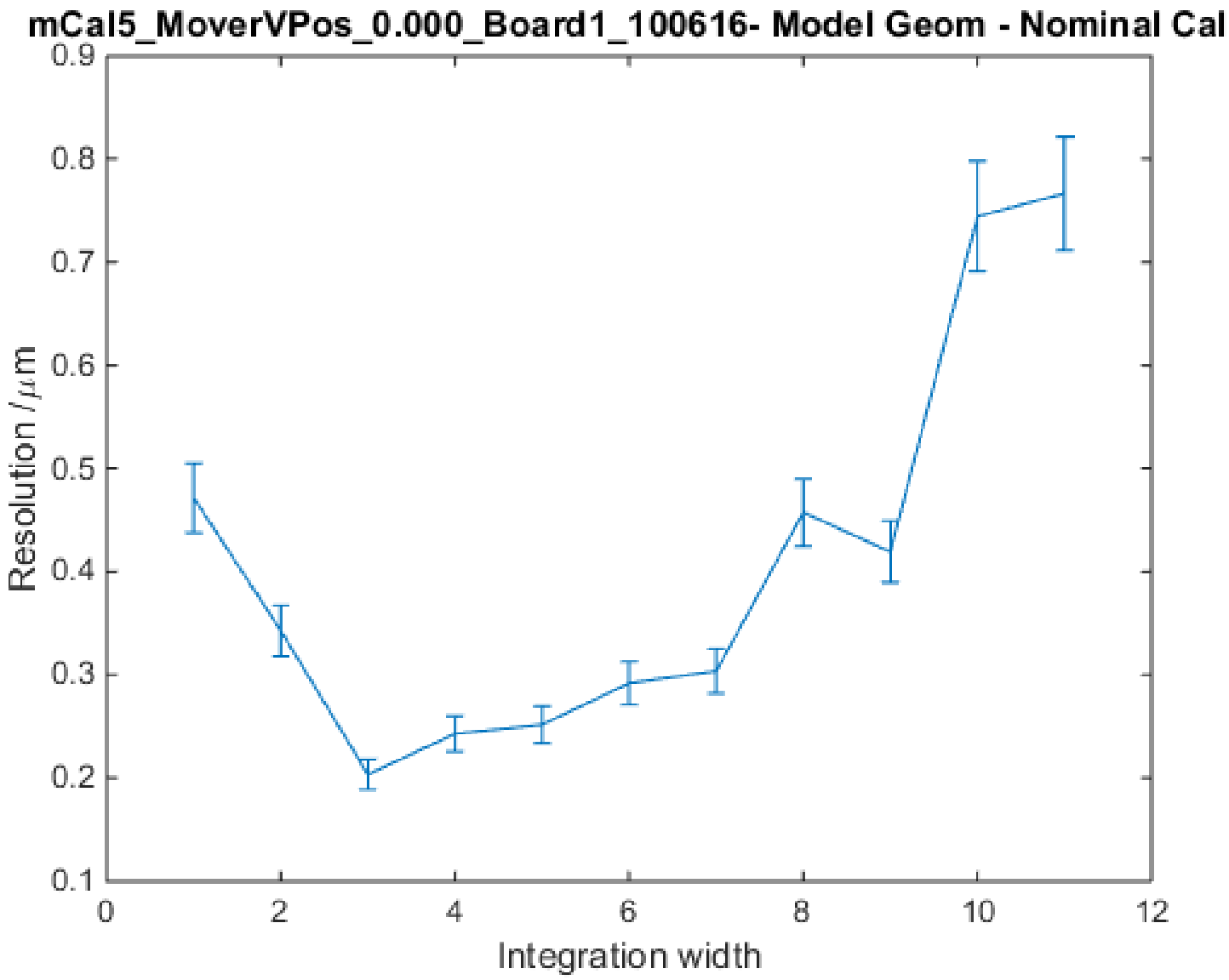
Nominal calibration constant = 0.005

Resolution with Integration

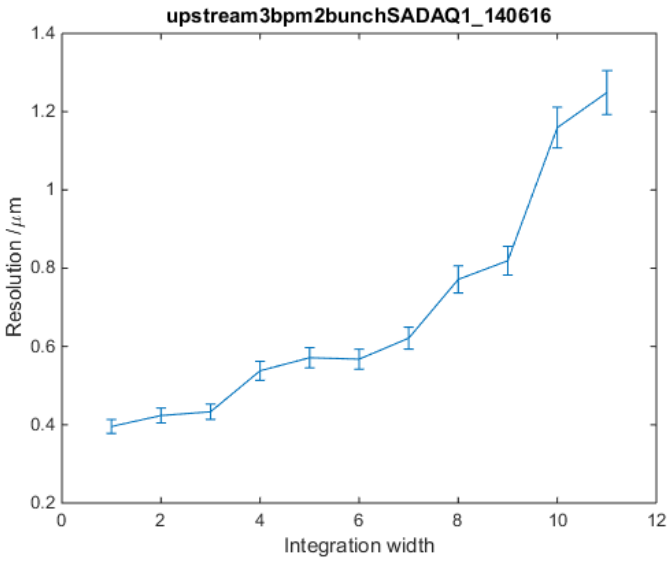
(mCal5_MoverVPos_0.000_Board1_100616 – HIGH CHARGE!!)

Integ width	Nominal calibration		Calculated calibration	
	Model*	Fit	Model*	Fit
1	0.47	0.42	0.44	0.40
		0.17		0.17
		0.30		0.29
3	0.20	0.17	0.18	0.16
		0.12		0.12
		0.17		0.16
5	0.25	0.21	0.21	0.19
		0.15		0.14
		0.21		0.19

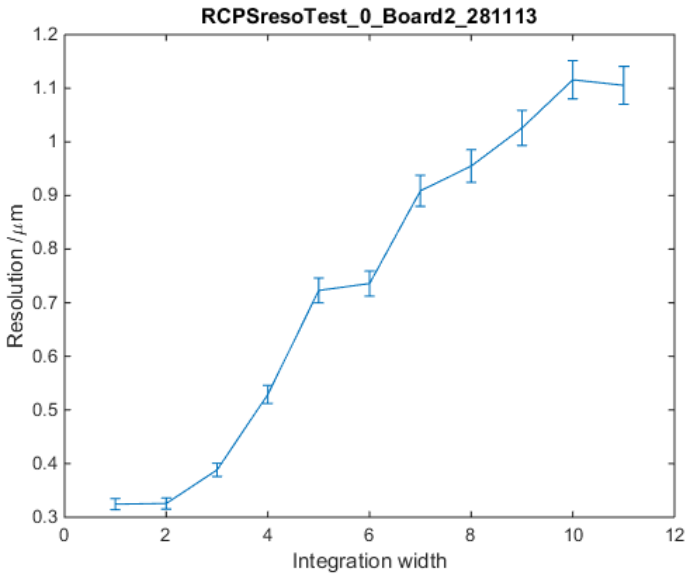
* NB: model does not correspond to the correct lattice for the dataset, but is used to compare


mCal5_MoverVPos_0.000_Board1_100616- Model Geom - Nominal Cal
Resolution /μm
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
2
4
6
8
10
12
Integration width

However, unfortunately …



Low charge data from week after taken with new (CTF) firmware and Standalone DAQ.



Classic 2015 BPM paper dataset. Taken with old FONT5 board #2.

Interpretation

- Clearly integration is having an effect for the high charge dataset, however it makes things worse for other datasets!
  - Second order effect, i.e. artefact of something else degrading the resolution?
  - E.g., integration compensates for source of noise which is artificially limiting the high charge data, eg sampling jitter? Perhaps low charge data is actually noise limited above this threshold.
- Gives some confidence that the underlying resolution is (would be) below 200 nm!
- Other (3 dB) effects:
  - Sampling jitter !
  - ADC noise (removal of 3 dB attenuators at output of processors)
- If we did want real-time integration, would require re-design of the upstream FB firmware
  - Always envisaged anyhow!

Firmware to-do List

- Sampling jitter (zeroth priority -> ~1 week)
  - Complete studies from 4-6 weeks ago
    - Correlation of jitter across ADC banks/channels
    - Stripped back firmware (removal of PLL and DCM-360)
    - Removal of 40 MHz (for ATF F/W) and replacement with derived clock (eg 357/11)
- CTF F/W (Highest priority! ~ 1 week -> 1 month)
  - Bugfixes to existing firmware
  - Averager/Combiner module(s)
- F/W mods for GM-FF (Doug) (~ 1 hour -> 2 days)
- New ATF2 firmware for IPFB (and upstream?) (low priority for now)
- Other ideas/issues (Nil priority – abandoned for now)
  - Ultra-high latency u/s firmware ??
  - ADC alignment monitor overhaul …
  - …

ATF2 IP(/u-s) FW

- Planned to do new module for IP-FB based on new firmware base for CTF
  - allow for multi-sample averaging (integration)
  - multiplexed FF-inputs (i.e. A,B -> C; A,C -> B; B,C -> A)
  - …
- Same firmware (configurable with `BUILD options) can be used upstream especially if eg no-PLL, no-40MHz, multi-sample integ needed
  - Possibly limited to 2 bunch operation initially!

CTF PFF F/W

- Bugfixes (Highest priority)
  - Fix overflow detection on output IIR filter [~1 day + ~> 1 day to remind myself what the problem is, how id all works etc]
  - Other low-level problems!! ??
- Averager/combiner module (~ 1-2 weeks - > 1 month)
  - Original plan to have a common module to do both
  - Average/combine factor 2,4,8
  - no longer sure about this …
  - Also, planned to use 300/400 MHz clock (3 GHz derivative) synchronous to sub-pulse freq.
    - Maybe can just use 357 MHz, and live with non-perfect correction at the ends of the pulse – should keep all timing aspects same
  - Minimal approach (High priority)
    - Just combiner module (maybe just factor 8 for simplicity)
    - Just average (x2) at the output – compensate for decimation.
  - Other idea
    - Subtract every other sample, ala slow drift correction?
    - Allow correction along the entire pulse(?), eg phase sag not seen by PFF correction
    - Relies on timing of correction being set very accurately
      - Would averaging help with this, i.e. (b-a)+(c-b) = (c-a)?

F/W for Doug (1hour -> 2 days)

- Have a new UART module operating on DigInA:
  - capable of any connection speed (75 Baud to 25 MBaud)!)
  - Configurable word width
  - Also, added two-byte little-endian decoder to send 2 x 8-bit bytes (= 14-bit value) with MSB representing the byte sequencing, though (I think) this is untested.
- Two options for 14(13)-bit operation
  - Either send single jumbo-byte (13/14 bit payload)
    - Should work straight-off-the-bat
  - Or, use the two-byte decoder
    - Preferred, but may require testing in lab!

New UART continued (lowest priority)

- For new (CTF) firmware made several changes to Ben's original UART module
  - Tidying code/removing unused functions
  - Enabling real-time switch between three Baud rates: 115200, 230400, 460800
  - Separating UART TX and RX into different modules (clocked on different domains – UART RX assumed to need to keep running even when 40 MHz clock disabled (sampling)
- But essentially still the old UART module(s) with absolutely no error checking
- Would like to update the main (DAQ) UART to new module including:
  - Fully configurable speed (already implemented). Real-time switching not routinely necessary (only used by me!)
  - Configurable as simplex (RX or TX), or full-duplex. N bits, parity (none,odd,even), N stop bits ..
  - Error checking
    - Detect framing, and RX/TX overrun errors
    - Clk16x? Framing errors, noise
    - Parity checking? Would not always want to use, due to overhead, just if suspect problem in data transmission
    - Would be better to append simple checksums to all RAM contents (e.g. 11 x DAQ RAMs, 1 x CR memory, 1x FB gain-LUT RAM)
- All not very much work, and some of it (at least some error checking) should be done, but low priority nonetheless!

DAQ issues

- ….