Beam Position Monitoring using MicroTCA for Diamond-II

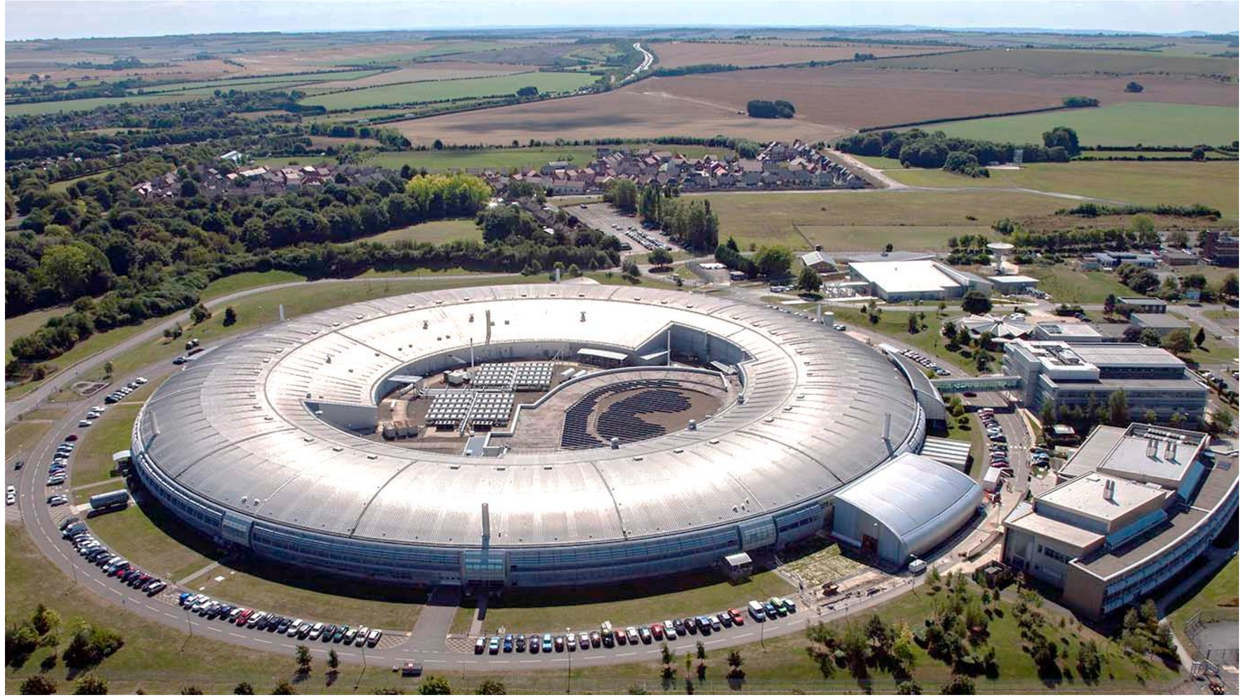
MTCAWS – DESY, Hamburg

Laurence Stant

December 7th, 2022

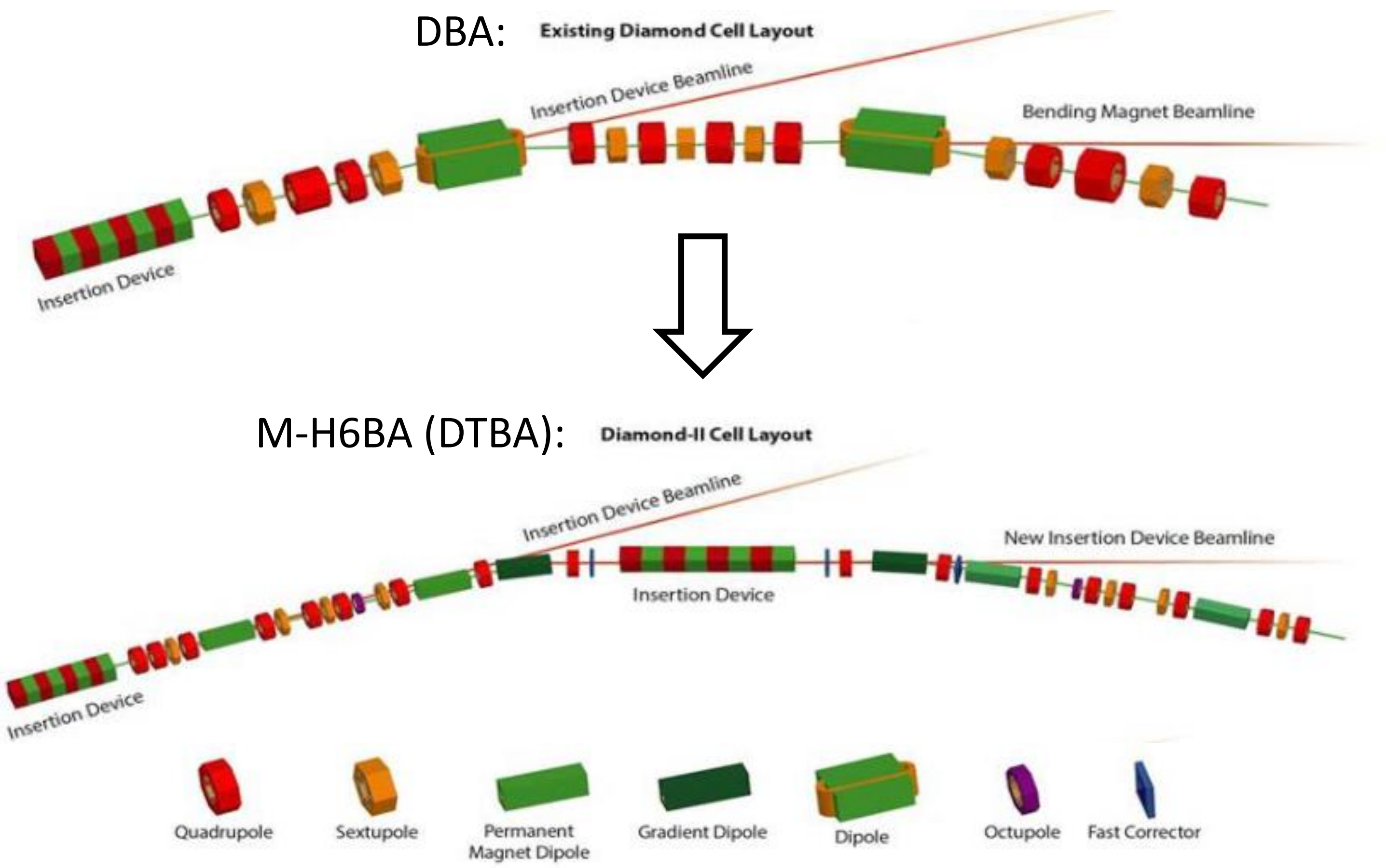
Agenda

- Diamond-II Upgrade
- BPM System Requirements
- BPM Cell Overview
- EBPM Design
- XBPM Design
- Communication Architecture
- Supporting Work
- Conclusion



Diamond-II Upgrade to 4th Gen. Light Source

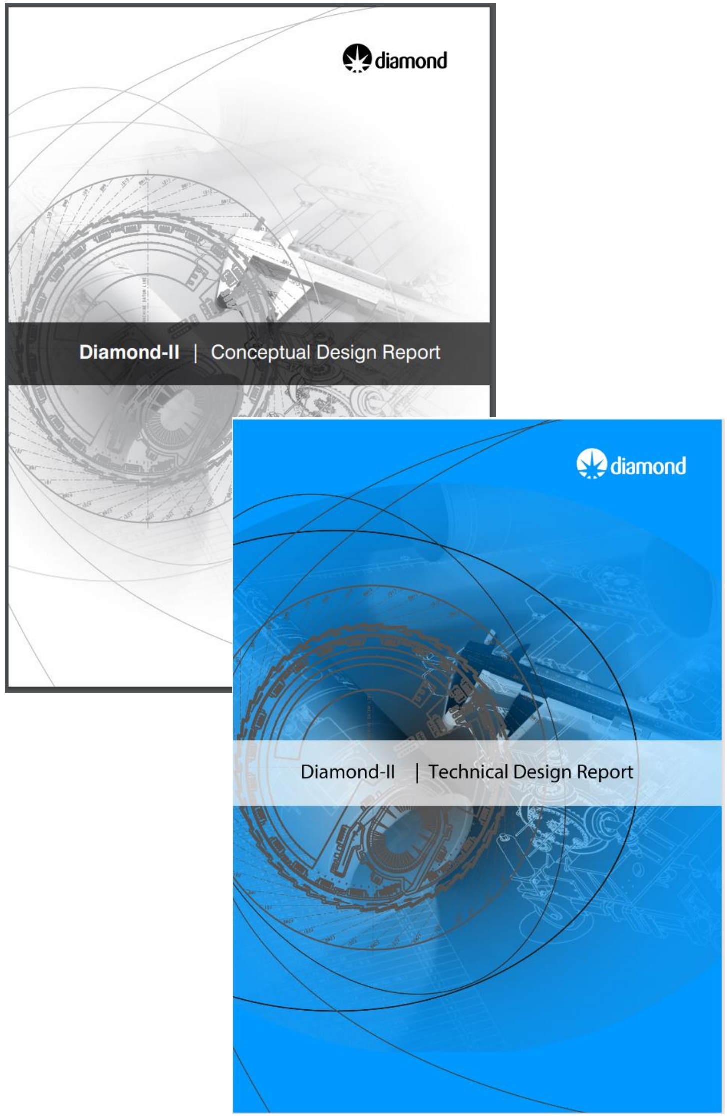
- Replacing entire machine at Diamond Light Source.
- Energy increased from 3 to 3.5 GeV.
- Horizontal emittance reduced from 2700 to 160 pm rad.
- RF @ 499.499 MHz
- Double Triple Bend Achromat cells.
- 24 new “mid-straights” available for ID positions: 5 new beamlines and upgrades to bending magnet beamlines.



BPM System Requirements

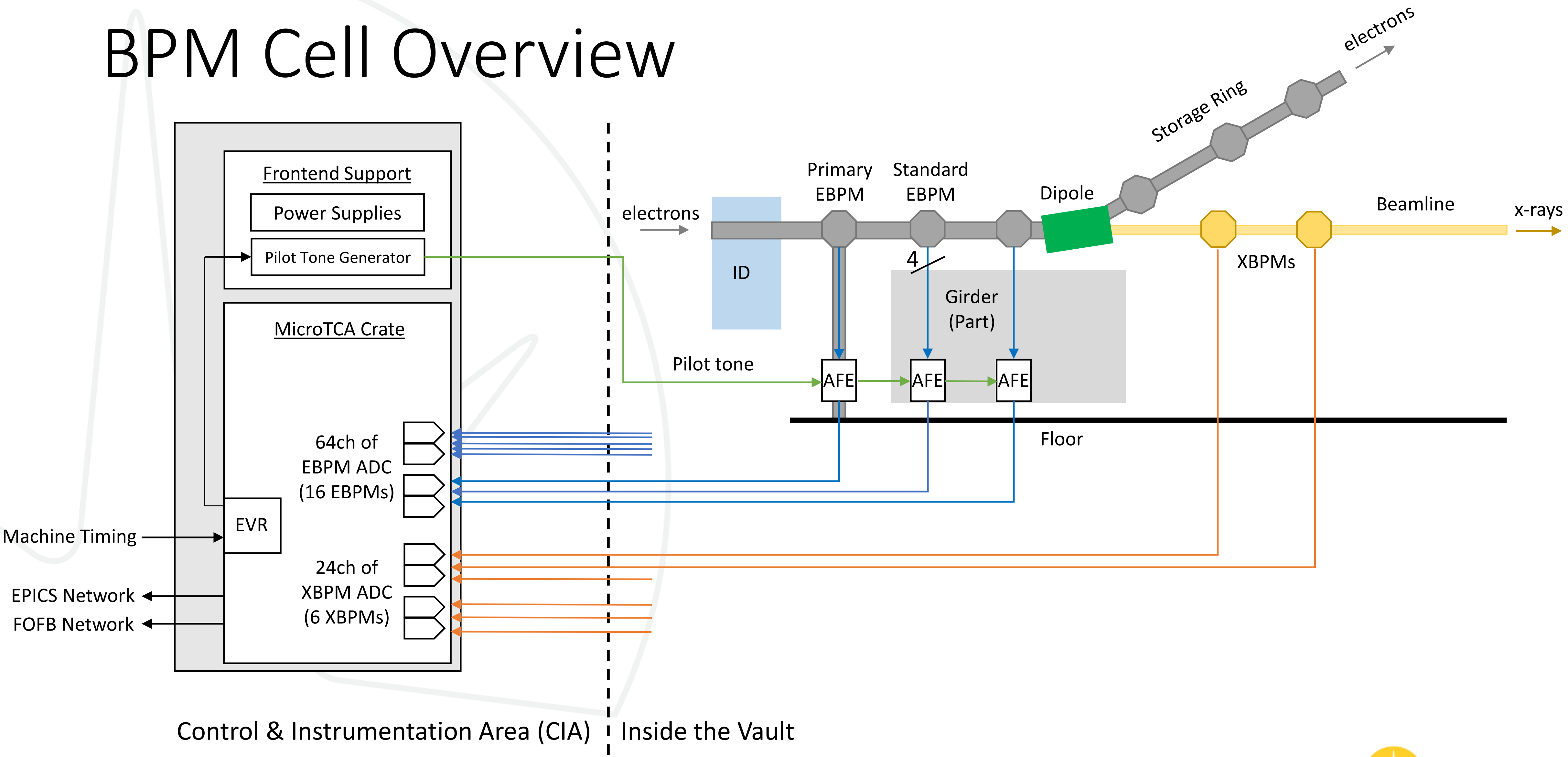
- Lower emittance requires improved FOFB.
- EBPMs and XBPMs included in FOFB.
- Data rate must increase from 10 to 100 kHz.
- 252 SR EBPMs (11/12 per cell).
- 60 XBPMs.
- Short-term motion (<1 second):
  - Commissioning (0.3 mA): <130 nm/√Hz
  - User beam (300 mA): <2 nm/√Hz
- Long-term motion (<1 wk): <1 µm pk—pk
- In-house design chosen due to sufficient technical resource and concern over legacy COTS capability (in 2020) with new data rate requirement.
- MicroTCA chosen for integrated gigabit backplane, COTS availability and longevity (15 years support from 2027).

https://www.diamond.ac.uk/dam/jcr:ec67b7e1-fb91-4a65-b1ce-f646490b564d/Diamond-II%20Conceptual%20Design%20Report.pdf



https://dlsltd.sharepoint.com/:b:/s/DiamondCommunications/EeZYURxeZQ9BsjSXqsWGd7kBbDWvOjFNAhjiD1mK-N7KIg?e=D9coh2

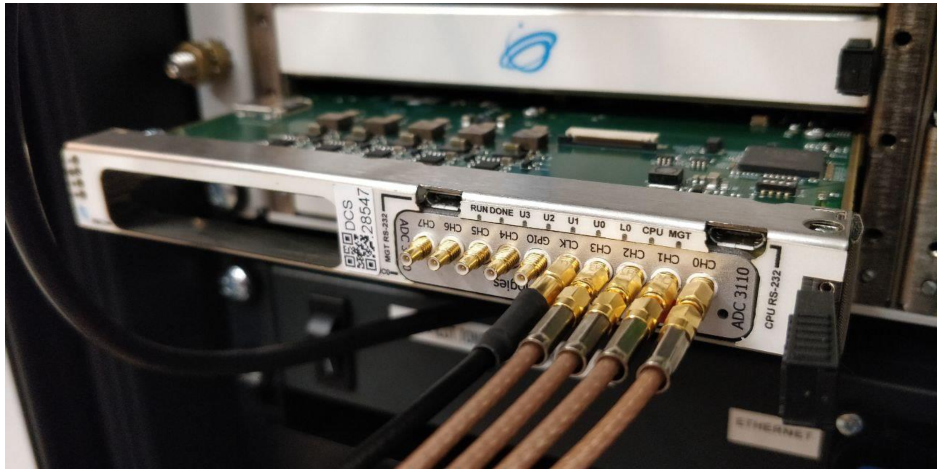
BPM Cell Overview



diamond

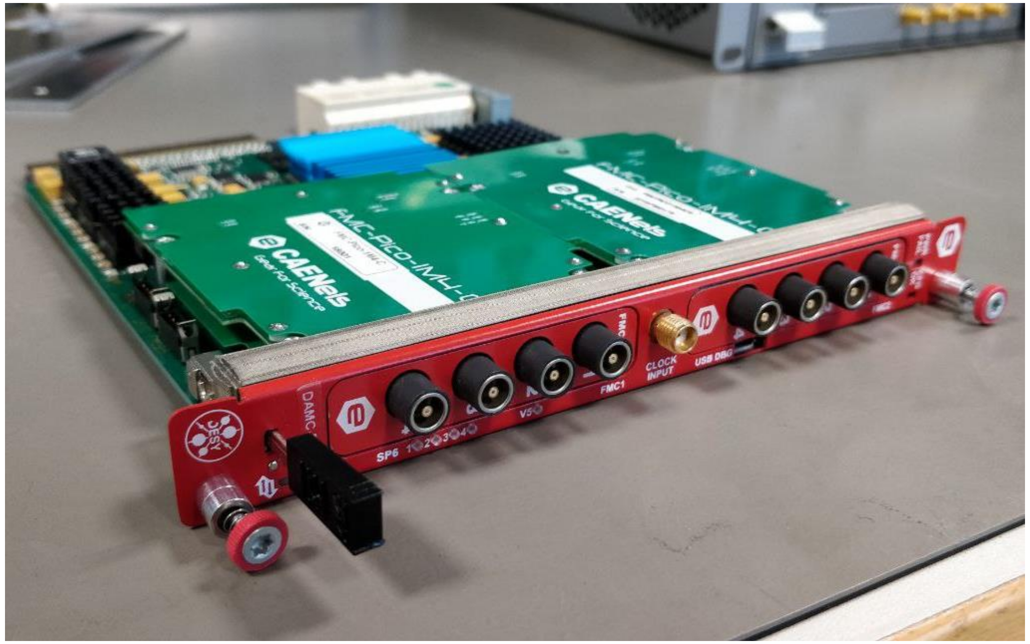
EBPM Design

- Button pickups in the storage ring generate 500 MHz waveforms which are amplified and filtered by an analogue front end and then digitised on FMC modules hosted on an AMC carrier.
- The front end also injects a pilot tone signal at a close frequency and this compensates out system gain and phase variations. This is extracted in the digital processing.
- We have planned to use IOxOS ADC3110 8ch 16-bit/250 Msps FMCs on a suitable dual AMC carrier.
- We are currently testing on a Vadatech AMC561 with a Virtex 7 690T carrier.
- We will be testing on a prototype IOxOS IFC_1412 carrier with Ultrascale+ FPGA.
- The high density of fast ADCs (8 per AMC) is sensitive to the carrier’s power implementation.



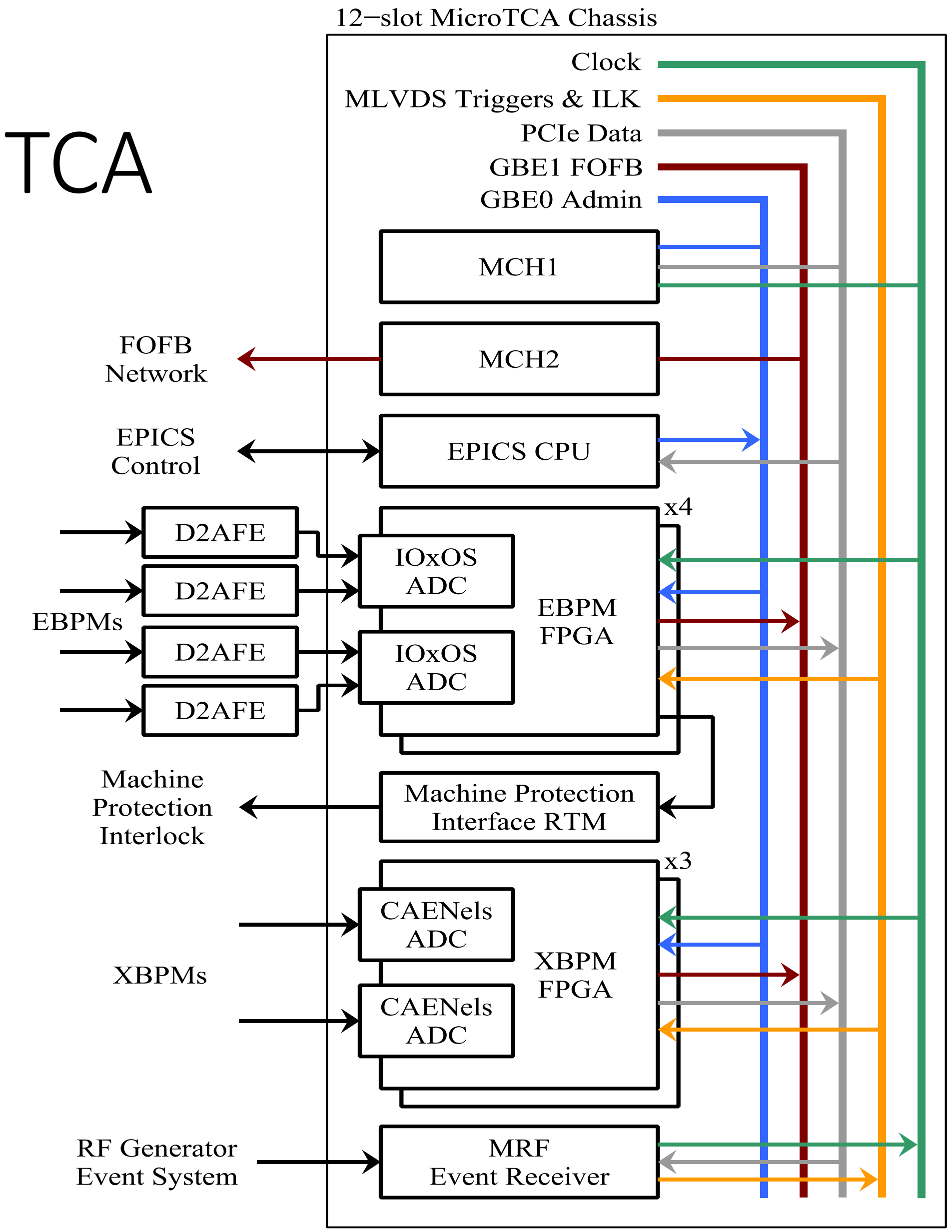
XBPM Design

- Electrical currents are induced in metallic blades placed close to the x-ray beam. These are connected to specialised digitizer FMCs and are hosted on an AMC carrier.
- We have planned to use CAENels FMC-PICO-1M4-C6 4-ch 20-bit 1 MSPS pico-ammeter FMCs.
- We are currently testing on a CAENels DAMC-FMC25 AMC carrier.
- We will be testing with the same IOxOS prototype carrier as the EBPMs for standardisation.



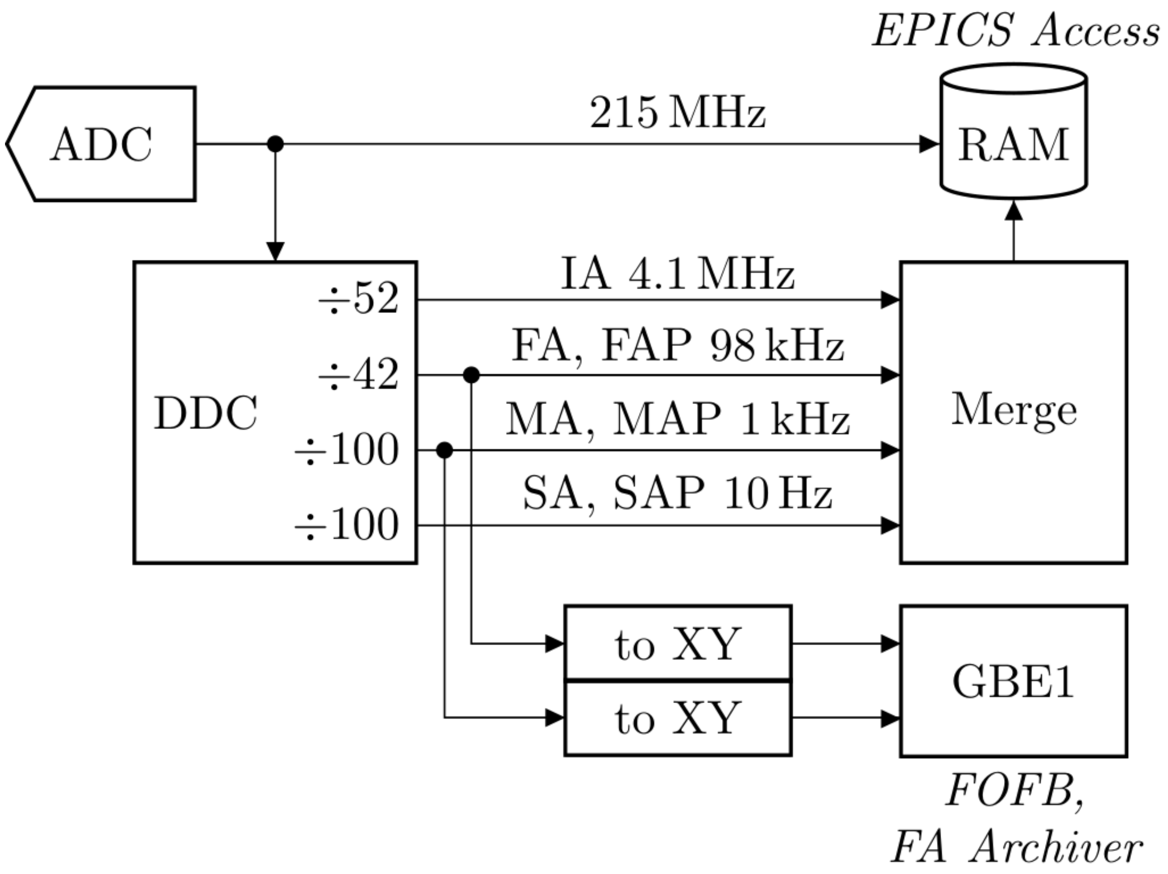
Communication Architecture - MicroTCA

- A single 12-slot crate handles an entire cell of EBPM and XBPM diagnostics.
- Four EBPM dual-FMC carrier cards.
- Up to three XBPM dual FMC carrier cards.
- Dual MCH: One for PCIe, one for FOFB GbE.
- Event receiver card, CPU card.
- Machine protection from custom Rear Transition Module (RTM).



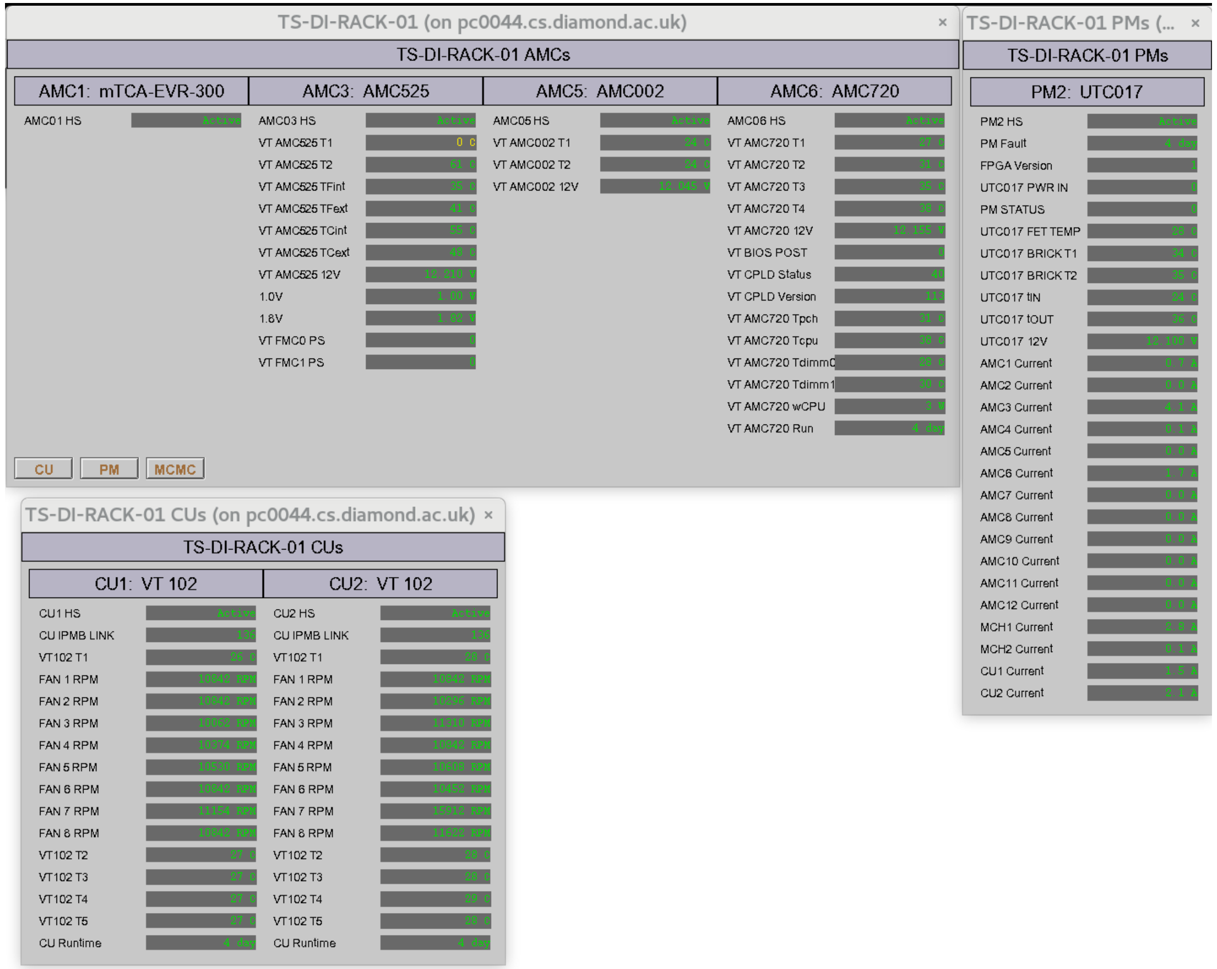
Communication Architecture - Streams

- 215 Msps ADC streamed to RAM.
- Downconverted with 70 MHz IF and then downsampled into streams:
  - 4.1 MHz intermediate acquisition
  - 98 kHz fast acquisition (+ pilot)
    Used for FOFB and fast archiver
  - 1 kHz medium acquisition (+ pilot)
    Used as a new archiver source
  - 10 Hz slow acquisition (+ pilot)



Supporting Work – EPICS MTCA Management

- We have developed IOC code to enumerate and poll FRU data via python-softioc and python-ipmi.
- Auto generates EDM screens (could extend for CSS).
- Easy to use, just give it an MCH IP address.
- This is available as a Github repository at: https://github.com/EmilioPeJu/epicsmonmtca



Supporting Work – OpenMMC STM32 Port

- For Diamond AMC capability we have been refactoring and adding to the CERN OpenMMC STM32 porting effort from Adam Wujek and Tomasz Wlostowksi.
- We have a release soon to be added as a branch of the upstream.
- We developed a Python MCMC (MCH) simulator which connects to an MMC under test via an Arduino for USB <> IPMB-L **and** hardware-in-the-loop simulation for regression testing of IO and interrupt/event handling.
- Code at: https://github.com/EmilioPeJu/MMCTester
- Use latest python-ipmi as we had to submit some fixes for better support!

Conclusion

- The Diamond-II upgrade involves a largely in-house BPM platform, both for EBPM and XBPMs.
- The MicroTCA platform has been chosen for data acquisition and processing due to the extensive COTS availability, established community and experience of good reliability from previous projects.
- Full cell deployment is now being tested for reliability and interoperability.
- More details in IBIC 22 paper: https://ibic2022.vrws.de/papers/mo3c2.pdf

Thanks

- With thanks to my co-authors:

  Michael Abbott, Chris Bloomer, Lorraine Bobb, Emilio Juarez-Perez, and Austin Rose.

- We would like to thank Dr. Guenther Rehm of Helmholtz-Zentrum Berlin for initial development work on this project.

- …and thank you for your attention!