

- EICROC1 with ZCU106 [TL]
  - issues I had using only the new lpGBT mode understood – **works** (milestone!)
  - currently the readout is super slow (0.5 Hz) – working on a faster scheme but even that can only reach perhaps 5-10 Hz
    - the reason is slow serial communication with the host PC and/or full 1024 pixels worth of data
    - **working on a solution using the embedded Zynq Xilinx CPU** [Prithwish, TL]
      - can reach 100's of Hz via ethernet
  - **EICROC1 has known issues for the “bad” version**
    - I2C read doesn't seem to work well (not really a problem)
    - **The 5 per-pixel setup registers can be written either for**
      - all pixels at the same time OR
      - 1 specific pixel at a time
    - but this is good enough for basic tests and to develop procedures, IMHO
  - **Potential issues with triggered data in the lpGBT mode is latency from the time of trigger through lpGBT to the ASIC**
    - need to measure this latency once we start using RBv1
    - NOTE: we can't use the legacy interface “TRGOUT” with lpGBT
  - Other measurements coming soon [Rachel, TL]
    - jitter of the 320 MHz clock: with ZCU106, with RBv1
- **Rachel** – Rice student at BNL for 2 months
  - understand the EICROC1 registers and their required values for different configurations and test modes
  - will work with Alex to transfer his EICROC0 register setups to EICROC1 (majority of the register bits are the same but in different registers)
    - also involved in calibration methods and data taking procedures w Alex
  - EICROC1 data interpretation (plots etc) w Alex & TL
  - various scope measurements (clock jitter, trigger latency) with Prithwish & TL
  - eventually involved in RBv1 lpGBT configuration and readout procedures w TL, others
- **OMEGA France [Christophe]**
  - last week shipped 60 “bad” EICROC1s to BNL
  - and also 2 more EICROC1 Testboards → **but we need to wirebond the ASIC at BNL**
    - **after which we plan to first test them with ZCU106**
  - **once this is done we will modify 1 of the Testboards for use with RBv1**