

XTRP Readout Errors

- The problem: The data in found in the XTRD bank disagrees with the simulation generated from the XFLD bank at low rate.
 - We have verified that the trigger data is good; the XTRD bank is simply getting the wrong information.
 - Bunch counter errors occur most frequently, followed by DIRAC bit errors and more rarely track word readout errors.
 - Errors are likely caused by a conflict between L1A & L2A or L1A & VME transactions.
 - We are trying to look at the timing of the reads and writes on error events to confirm these conflicts cause the errors.
 - We are trying to examine the problem in L2 torture test runs.

Testing Thus Far

- Testing which looked at the rate of BC errors in the data boards indicate that the error rate is directly proportional to the L1 accept rate.
- Beam test and L2 torture data indicate that the 3TT board also has bunch counter errors.
- Mike Kasten has written code to capture bunch counter and L2 buffer info on every L1A and L2R from the 3TT.
 - In the runs we have taken since we began using this code have had no readout errors in either the 3TT or the data boards. We don't know why.

New Problem

- Cheng-Ju recently discovered a new problem in the L2 tracklist received from XTRP.
- There is an extra word with nonsense global phi and pt bits (0x1ffc1f) that appears in the L2 readout list at a rate of ~ 2 per 10^6 events. It is not clear yet if this is an XTRP or a Tracklist board problem. Thus far, only 2 events have been found with this error.
- Question: Does L2 treat this as a non-track? $p_T(\text{bin})=127$ is undefined.

Plans for Future testing

- Try taking more L2 torture test runs with different trigger rates and trigger tables to see if the readout errors will return.
- Scan more events to look for more nonsense L2 Tracklist words.

Other XTRP Progress

- **ϕ -gap triggers:** New map code fixes even/odd wedge error. Anyes putting fix into coldstart. Can have high- p_T CMNP ϕ -gap trigger after shutdown.
- **IMU maps:** Tighten maps to turn-off bogus adjacent wedge bits (see ϕ -gap problem). Camille checked it with IMU $W \rightarrow \mu$ and sees $>99\%$ efficiency. Better defined acceptance, will reduce trigger rate at L1.
- **XTRPSim:**
 - Handles ϕ -gap fixes (backward compatible)
 - Handles new IMU maps (need to put in trigger table flag)
 - Now generates XTRD bank from TL2D bank, can seamlessly run XTRPSim regardless of ROL.