Trigger + DAQ discussion

A. Sarti & G. Traini

DIPARTIMENTO DI SCIENZE
DI BASE E APPLICATE
PER L'INGEGNERIA

SAPIENZA
UNIVERSITÀ DI ROMA

Learning from the past (I)

➡ In view of dec data taking we need to carefully review what 'went wrong' the last time in GSI, to be prepared…

- Next slides contain what (to my limited knowledge .. and with the help of Giacomo) we recall that was problematic last time..

- please add whatever is missing at the end of the presentation during the discussion.

➡ Key ingredients present in GSI 2019 data taking: SC, TW, BM & VTX.

➡ Starting from TW: we lost events (many!) since from a given event number we experienced data transfer problems from the WaveDAQ boards to the event builder..

- Problem of waveDAQ event building/data transmission btw boards and workstation… Intermediate board is now available? … solved now?

Learning from the past (II)

➡ BM. To reconstruct the tracks we need to minimise the jitter affecting the time measurements btw the SC and the BM signals. In Isi 2019 we had a big problem of signal re-sync that prevented to use the BM information with proper time resolution (that means poor spatial resolution!)

- Wave Dream 'gives' signal synced with its clock. Trigger is sent to the 2495 board that uses it.. Re-syncs the signal with its clock and send it to the detectors. So we have two independent jitters… (clock WD + 2495)…
- Sync TDC is hence affected by both. This spoils completely the time resolution. Needs to be changed, to use BM with correct time resolution.
- How can we prevent this?

Learning from the past (III)

➡ VTX: we lost many events..

– problems happened both at detector and at event building level.

– What is the status of the fix?

– how can we 'build events' being sure that we can be able to re-assmble them even in a 'post processing' analysis? Which flag/variable we want to use: use trig id? time?

Future tests

➡ MSD + CALO integration, are any problems foreseen?

- – + 1 wave dream.. additional data.. it is a problem? shall we test before the integration? how/when?

➡ Neutrons... will be in the 'calo' WD, so should be ok...

- – we need to foresee a dedicate trigger line..

Planning the future frag trigger..

➡ We will have the chance to test the trigger strategy, implementation and efficiency measurements during the data taking.. everything needs to be planned carefully and in due time.

➡ HW:

– We should include the handling at FPGA level of the info from Calo as well..

– strategies are implemented @ firmware level.. doing it online is not easy..we need to arrive prepared. (which tests/strategies, which info we want..)

➡ Basic ingredients: SC, TW, CALO

– Discriminated signals from the detectors are available.. to use them the timing of signals needs to be accounted for in the implementation inside FPGA.

– **Length of cables is crucial… design phase … people should be aware since the beginning…**

What we want to test?

➡ Severa triggers to be tested are already known..

- Min bias trigger: majority of SC.

- The prescale implementation and tuning.

- Frag1: Veto (central bars) + majority + at least 1 in front 1 in rear.

- Frag2: shall we add CALO (as veto)

- Check CALO in different positions to mimic frag with calo?

- Neutrons..

➡ For each condition we need to decide: how many events, which condition, which detector setup. this is needed to plan with due time the data taking and time needed in the global schedule.

➡ We need all HW trigger signals in the output… to measure the trigger efficiencies both on data and to perform comparisons with MC.

What can we measure?

➡ Data/MC validation.

- Check of how robust are our strategies to compute trigger efficiencies and to assign a systematic uncertainty to them.
- Angelica will implement the geometry we want to test inside the MC and start some simulations to have numbers to start with…

➡ How we measure it?

- raw hits distributions to get mean rates…
- global reconstruction will be probably needed only to clean the sample and evaluate the bkg impact.

➡ Veto efficiency

- Use calo info…